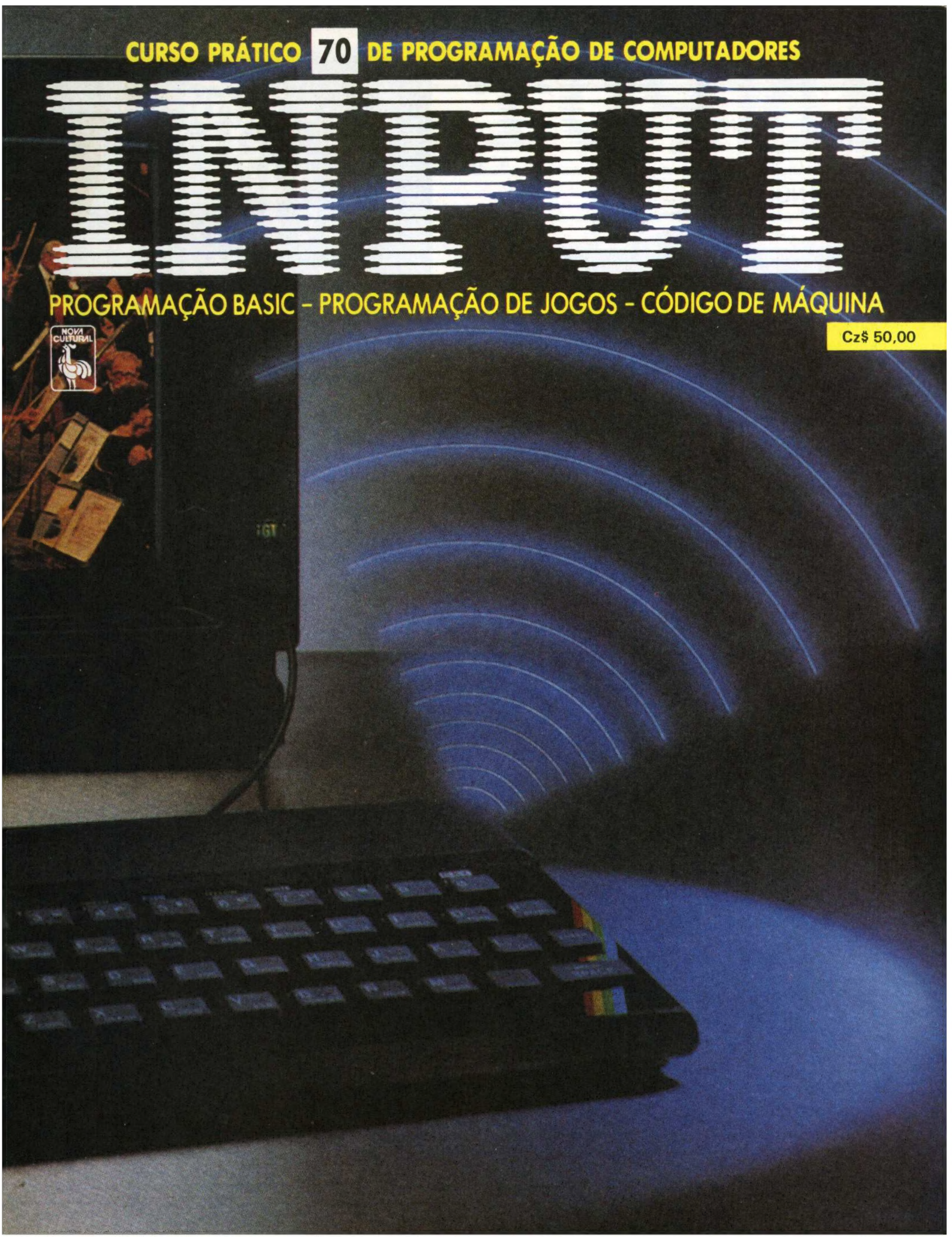
CURSO PRÁTICO 70 DE PROGRAMAÇÃO DE COMPUTADORES

INPUT

PROGRAMAÇÃO BASIC - PROGRAMAÇÃO DE JOGOS - CÓDIGO DE MÁQUINA

NOVA CULTURAL

Cz$ 50,00



INPUT

Vol. 5 Nº 70

NESTE NÚMERO

SOFTWARE

PROCESSADORES DE TEXTOS

Correção eletrônica. Limpeza do texto na tela. Hardware necessário. Impressão 1381

APLICAÇÕES

DESENHO ARQUITETÔNICO (2)

Complemento das listagens. Como gravar, carregar e imprimir o projeto. Uso do programa 1386

PROGRAMAÇÃO BASIC

DESENHOS EM PERSPECTIVA

O que é perspectiva. Ponto de fuga e ponto de vista. Diminuição de tamanho. Técnicas de sombreamento. Pontilhado 1391

PROGRAMAÇÃO BASIC

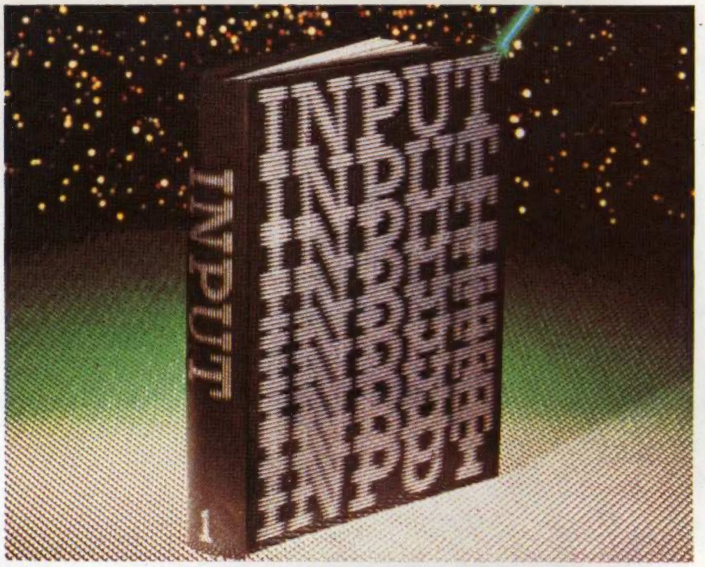
FORMATAÇÃO DE TELAS

Rotina para entrada de dados no Apple e no TK-2000. Como funciona. Aperfeiçoamentos 1396

APLICAÇÕES

UM EDITOR MUSICAL(1)

O que é um editor musical. Digitação das notas. Execução de melodias no teclado..... 1398



PLANO DA OBRA
INPUT é uma obra editada em fascículos semanais, e cada conjunto de 15 fascículos compõe um volume. A capa para encadernação de cada volume estará à venda oportunamente.

FÉRIAS, VIAGENS, MUDANÇAS...
NÃO FIQUE COM A COLEÇÃO INCOMPLETA
Se você está saindo de férias, pretende viajar ou vai se ausentar por algum tempo, avise antecipadamente seu jornaleiro. Ele pode guardar os seus fascículos enquanto você estiver fora. Se, por qualquer motivo, você perdeu alguns números, peça-os também a seu jornaleiro, ou entre em contato com nossa Distribuidora:

1. **Pessoalmente** — Em *São Paulo,* os endereços são: rua Brigadeiro Tobias, 773, Centro; av. Industrial, 117, Santo André. No *Rio de Janeiro,* av. Mem de Sá, 191/193, Centro.
2. **Por carta** — Envie para:
   DINAP — Distribuidora Nacional de Publicações
   Números Atrasados
   Estrada Velha de Osasco, 132 — Jardim Teresa
   CEP 06040 — Osasco — SP
3. **Por telex** — Utilize o nº (011) 33 670 DNAP.

Em **Portugal**, os pedidos devem ser feitos à Distribuidora Jardim de Publicações Lda. — Qta. Pau Varais, Azinhaga de Fetais, 2685, Camarate, Lisboa; Apartado 57; Telex 43 069 JARLIS P.
Atenção: Após seis meses do encerramento da coleção, o atendimento dos pedidos dependerá da disponibilidade do estoque.
Obs.: Quando pedir livros, mencione sempre o título e/ou autor da obra, além do número da edição.

COLABORE CONOSCO
Encaminhe seus comentários, críticas, sugestões ou reclamações ao
SERVIÇO DE ATENDIMENTO AO LEITOR
Caixa Postal 9 442, São Paulo — SP — CEP 01051.

EDITOR
RICHARD CIVITA

NOVA CULTURAL

Presidente
Flávio Barros Pinto
Diretoria
Carmo Chagas, Iara Rodrigues,
Pierluigi Bracco, Plácido Nicoletto,
Roberto Silveira, Shozi Ikeda,
Sônia Carvalho

REDAÇÃO
Diretor Editorial: Carmo Chagas
Editores Executivos:
Antonio José Filho, Berta Sztark Amar
Editor Chefe: Paulo de Almeida
Editoras Assistentes: Ana Lúcia B. de Lucena,
Marisa Soares de Andrade
Chefe de Arte: Carlos Luiz Batista

Assistentes de Arte: Dagmar Bastos Sampaio,
Grace Alonso Arruda, Monica Lenardon Corradi

Colaboradores
Consultor Editorial Responsável:
Dr. Renato M. E. Sabbatini
(Diretor do Núcleo de Informática Biomédica da Universidade Estadual de Campinas)
Execução Editorial: DATAQUEST Assessoria em Informática Ltda., Campinas, SP
Tradução, adaptação, programação e redação:
Abílio Pedro Neto, Aluísio J. Dornellas de Barros,
Marcelo R. Pires Therezo, Marcos Huascar Velasco,
Raul Neder Porrelli, Ricardo J. P. de Aquino Pereira
Coordenação Geral: Rejane Felizatti Sabbatini

COMERCIAL
Diretor Comercial: Roberto Silveira
Gerente Comercial: Flávio Maculan
Gerente de Circulação: Denise Mozol
Gerente de Propaganda e Publicidade: José Carlos Madio
Gerente de Pesquisa e Análise de Mercado:
Wagner M. P. Nabuco de Araújo

CLC

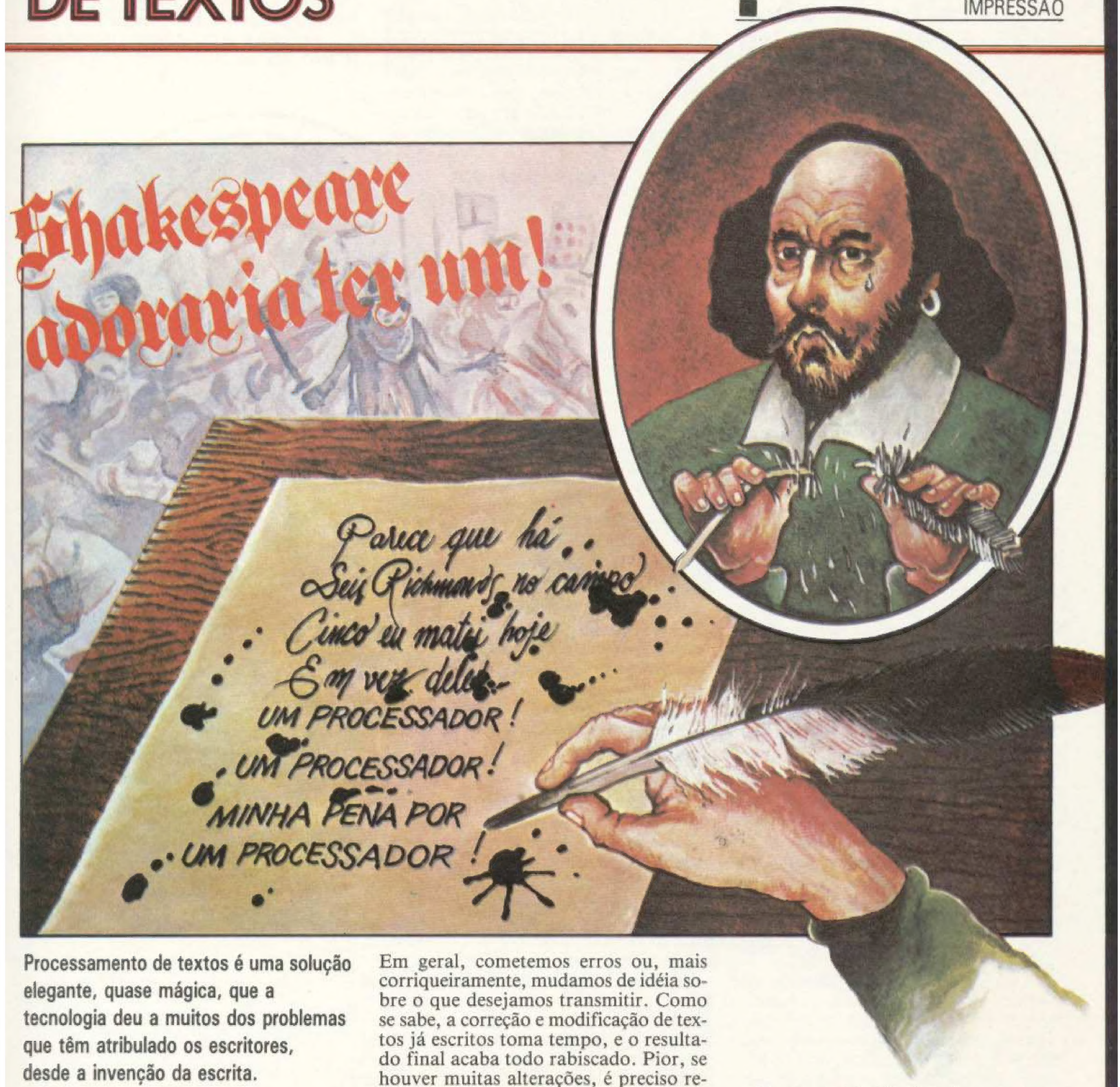
A Editora Nova Cultural Ltda. é uma empresa do Grupo CLC — Comunicações, Lazer e Cultura
Presidente: Richard Civita
Diretoria: Flávio Barros Pinto, João Gomez,
Menahen M. Politi, Renê C. X. Santos,
Stélio Alves Campos

© Marshall Cavendish Limited 1984/85.
© Editora Nova Cultural Ltda., São Paulo, Brasil, 1986.
Edição organizada pela Editora Nova Cultural Ltda.
Av. Brigadeiro Faria Lima, nº 2000 - 3º andar
CEP 01452 - São Paulo - SP - Brasil
(Artigo 15 da Lei 5 988, de 14/12/1973).

Esta obra foi composta na AM Produções Gráficas Ltda. e impressa na Divisão Gráfica da Editora Abril S.A.

PROCESSADORES DE TEXTOS

- CORREÇÃO ELETRÔNICA
- LIMPEZA DO TEXTO NA TELA
- HARDWARE NECESSÁRIO
- MEMÓRIA
- IMPRESSÃO



Processamento de textos é uma solução elegante, quase mágica, que a tecnologia deu a muitos dos problemas que têm atribulado os escritores, desde a invenção da escrita.

É quase impossível produzir um texto perfeito logo na primeira tentativa. Em geral, cometemos erros ou, mais corriqueiramente, mudamos de idéia sobre o que desejamos transmitir. Como se sabe, a correção e modificação de textos já escritos toma tempo, e o resultado final acaba todo rabiscado. Pior, se houver muitas alterações, é preciso rebater ou reescrever todo o documento, e novos erros podem aparecer. Quem já experimentou escrever conhece bem esse torturante processo.

O processamento de textos por computador apresenta a vantagem imediata de facilitar a execução de alterações, tornando desnecessária a cansativa ta-

refa de "passar a limpo" o trabalho feito. Isso resulta em tremenda economia de tempo, o que é de utilidade mesmo para aqueles que não escrevem por necessidade ou profissão.

Já publicamos em *INPUT*, no artigo da página 576, um programa que coloca à disposição do usuário os recursos básicos de edição de texto. Se você tem interesse em adquirir um processador mais potente, convém conhecer mais detalhadamente suas possibilidades.

ADEUS, BORRACHA!

Processar um texto significa simplesmente digitá-lo no teclado de um computador e armazená-lo na memória. Portanto, usando um software adequado, você poderá fazer modificações e correções com rapidez e eficiência, e em seguida proceder à impressão. Se o texto for armazenado em fita ou disquete, será possível carregá-lo novamente, sempre que necessário, para outras modificações.

Como vimos no artigo já mencionado, com um programa elementar de processamento de texto, o usuário efetua as alterações "passeando" um cursor (geralmente um pequeno retângulo iluminado) sobre a tela, até a parte a ser modificada. Escreve então sobre o texto que ali se encontra. Este é apagado da memória e substituído pelo novo texto — sem rabiscos e sem complicações.

Inserir letras, palavras ou frases no meio de um parágrafo também é uma operação simples. Para realizá-la, recorre-se à função de inserção dos processadores de texto. Por meio dela, podem-se introduzir alguns espaços em branco e depois escrever sobre eles. O processo mais comum, porém, é simplesmente escrever o que se quer inserir, deixando por conta do programa rearranjar automaticamente o restante do texto, para dar lugar ao novo.

Estes são apenas dois dos numerosos recursos disponíveis em um processador de texto. A potência e sofisticação desse software ficam evidentes quando se consideram os diversos "truques eletrônicos" de que é capaz para auxiliar quem está escrevendo.

As funções de inserção e modificação de textos são chamadas *recursos de edição*. Entre estes incluem-se também a busca e a substituição de caracteres no texto. A função de busca permite a localização automática de uma seqüência específica de caracteres — uma palavra, ou mesmo uma frase — em uma parte ou em todo o texto. Suponhamos que, tendo digitado um texto muito longo, queiramos fazer uma modificação em certo parágrafo. Não nos lembramos onde ele está: sabemos apenas que contém as palavras *espiral inflacionária*. A função de busca nos ajudará a descobri-las rapidamente.

A função de substituição é mais poderosa ainda. Ela efetua a busca de uma seqüência determinada de caracteres por todo o texto, substituindo-a sempre que a encontrar por uma segunda seqüência. Isso é útil quando grafamos incorretamente uma palavra qualquer em vários pontos do texto, ou quando queremos trocar uma expressão empregada com freqüência por outra de significado mais preciso. Com a função de substituição, podemos também recorrer a abreviaturas ou códigos na digitação de um texto. Por exemplo, se vamos escrever diversas vezes *programa amistoso para o usuário* no texto, bastará digitar um asterisco, ou qualquer outro símbolo, em cada um dos pontos em que a frase deve aparecer, usando posteriormente a função de substituição.

HABILIDADES ESPECIAIS

Com o processador de texto, não é preciso ser um datilógrafo habilidoso para conseguir bons resultados. Mesmo que você só consiga datilografar com dois dedos, observará um aumento considerável na sua velocidade de trabalho, pois poderá avançar "a todo vapor", sem se preocupar com erros ou efeitos estéticos.

Se sua ortografia deixa a desejar, procure utilizar um processador de texto dotado de corretor ortográfico (*spell-checker*, em inglês). Ele percorre o seu texto "olhando" cada palavra, e comparando-a com um dicionário interno padrão (armazenado em disco). Palavras não reconhecidas — por terem erro ortográfico, ou, simplesmente, por não constarem do dicionário — são assinaladas. Neste ponto, dependendo do programa, você tem três opções: deixar a palavra indicada como está, corrigi-la manualmente, ou permitir que o programa a substitua pela versão correta (muitos programas mostram uma lista das palavras que poderiam substituir a versão assinalada no texto).

FORMATAÇÃO

Os processadores de texto mais potentes dispõem de uma série de recursos adicionais, que permitem executar o que chamamos de *formatação do texto*. A formatação tem por objetivo produzir um texto final arranjado segundo certo padrão estético, independente da manei-

ra como foi digitado. Os recursos de formatação usualmente permitem especificar o espaçamento entre linhas, o número de linhas por página, o número de toques por linha, o tamanho da folha onde vai ser impresso o texto, a numeração das páginas, cabeçalho etc. A função mais importante da formatação, porém (e a mais difícil de se realizar manualmente), é a justificação das margens — ou seja, a definição do alinhamento lateral das palavras. A seção de formatação dos processadores de texto realiza essa tarefa automaticamente. Muitos deles chegam à sofisticação de separar as sílabas das palavras, ao final de uma linha, conforme as regras do idioma.

Quanto à formatação, existem três tipos de processadores de texto. Todos eles se encarregam das mudanças de linha, permitindo ao usuário datilografar o texto de um parágrafo continuamente. Em geral, a tecla <**ENTER**> precisa ser pressionada apenas ao final de um parágrafo.

No primeiro tipo de processador de texto, quando uma palavra que está sendo datilografada não cabe na linha, o cursor passa para a linha seguinte do vídeo, sem se preocupar em dividi-la segundo as regras gramaticais. O texto fica difícil de ser lido na tela.

Nessa situação, o segundo tipo de processador de texto transporta a palavra inteira para a linha de baixo (método denominado *word-wrap*, em inglês). A leitura no vídeo torna-se bem mais fácil, mas a margem esquerda do texto fica irregular, o que no Brasil recebe o nome de *margem americana*.

O terceiro tipo de processador ajusta a margem esquerda na tela, de modo a proporcionar um alinhamento. Como já mencionamos, alguns sistemas chegam a *hifenar* automaticamente as palavras que se dividem ao final da linha, o que melhora consideravelmente a aparência do texto. Esse processador é denominado *WYSWYG* (do inglês *What You See What You Get*). Com ele, o usuário tem oportunidade de acompanhar na tela — à medida que vai escrevendo — a produção do texto final.

O tamanho do texto, evidentemente, não é limitado ao tamanho da tela. A maioria dos processadores de texto permite "rolar" um texto para cima ou para baixo, ou, ainda, para a esquerda ou para a direita, de modo que a tela funcione como uma "janela" que mostra parte do texto. Nos micros pessoais, a janela tem, em geral, vinte linhas por quarenta colunas.

REALCE O SEU TEXTO

Os processadores de texto oferecem diversas opções de realce de seções do texto, como negrito, texto inverso (caracteres brancos sobre fundo negro, na impressão), letras de tamanhos ou formatos diferentes (itálico, dupla largura etc.). Esses realces podem aparecer na tela e na impressora, ou apenas na impressora. Nesse caso, caracteres especiais de demarcação assinalam, na tela, o lugar onde se inicia e termina uma função de realce.

COMANDOS DE CONTROLE

Os programas mais simples de processamento são os *editores de texto*, com as funções básicas de inserção e modificação. Um editor se transforma em um processador de palavras com o acréscimo de diversas outras funções, que podem ser utilizadas pelo usuário no momento em que necessitar. A maneira de ativar essas funções varia de programa para programa, e é um elemento importante na *ergonomia* (facilidade de uso) do processador.

As diretivas de controle dos recursos

...E ORWELL NÃO PODIA ESPERAR ATÉ 1984

do programa são entradas pelo teclado, num processo tão fácil quanto o da entrada de texto. Os *comandos funcionais* consistem normalmente de teclas especiais, ou seqüências de teclas. Entre as mais comuns, incluem-se as de retrocesso, as de controle dos cursores, <ENTER> etc. Bons comandos de edição são essenciais para facilitar o uso do programa. Em alguns casos, as seqüências de comando são muito complicadas e difíceis de memorizar — o que torna o programa pouco conveniente para o usuário ocasional.

Além desses, o programa possui comandos de controle de formatação cujo efeito só se observa, em geral, no momento da impressão. Nos programas WYSWYG, eles são "embutidos" no texto, sob a forma de caracteres invisíveis de controle. O vídeo responde a esses comandos de maneira bem semelhante ao que uma impressora responderia; isso possibilita que o usuário tenha uma visão prévia de como o documento ficará quando pronto.

Nos demais programas, os comandos de formatação são colocados explicitamente no texto, precedidos por sinais especiais. Por exemplo, um comando como:

.ME 20

introduzido no meio de um texto, indicaria ao processador que, naquele local, a margem esquerda deve ser modificada para vinte colunas.

CARTAS MÚLTIPLAS

Outro recurso de programas mais sofisticados de processamento de texto é a *mala direta* — a impressão de cartas múltiplas, endereçadas de forma personalizada a várias pessoas (veja o artigo da página 17).

Uma das maneiras mais simples de fazer isso consiste em escrever uma carta padronizada, com o auxílio do pro-

cessador de texto, e depois armazená-la em fita ou disco.

Para enviar essa carta a diferentes destinatários, bastará carregar o texto da mesma na memória e, utilizando as funções de edição, inserir nome, endereço e outros dados nos pontos apropriados, imprimindo-a a seguir.

Uma alternativa mais elaborada é o uso de um outro arquivo de textos — denominado *máscara* — contendo, por exemplo, o nome e o endereço de todas as pessoas para as quais se quer mandar uma carta. No texto da carta, por sua vez, introduzem-se caracteres especiais, que assinalam onde devem entrar os dados da máscara. Ao se fazer a impressão, o programa produzirá uma carta para cada destinatário, automaticamente, colocando os dados nos pontos certos do texto.

Um sistema de produção de cartas múltiplas tem variadas aplicações tanto domésticas (o envio de cartões de Natal personalizados, por exemplo), quanto profissionais (propaganda de um produto ou serviço, cartas circulares, correspondências para orçamentos e assim por diante).

ESCOLHA DO HARDWARE

Teoricamente, qualquer microcomputador pode ser usado para processamento de texto, desde que haja um programa adequado para ele (na série de artigos iniciada na página 576, apresentamos um programa completo de processamento de texto para a maioria das linhas cobertas por *INPUT*). É óbvio, porém, que não tem sentido fazer processamento de texto se não se conta com uma impressora. Além disso, convém lembrar que, na prática, existem algumas limitações impostas pelo hardware.

Para começar, há a questão do teclado. A maioria dos micros compatíveis com o ZX-81, por exemplo, possui um teclado inadequado para datilografia rápida de textos, não dispondo de letras minúsculas, nem de sinais de acentuação. Portanto, ao processar textos com essas máquinas, não se pode esperar um resultado de bom nível.

A linha Spectrum original (TK-90X, no Brasil) também tem um teclado sofrível, embora ofereça a alternativa de usar letras minúsculas e algum tipo de acentuação. O modelo TK-95 e os teclados semiprofissionais acessórios, que podem ser comprados para o Spectrum, melhoram bastante a situação.

Os computadores que possuem teclado semelhante ao de uma máquina de escrever — como o Apple, o TK-2000, o TRS-80, o TRS-Color II e o MSX — são mais adequados quanto a esse aspecto. Há, porém, uma grande variação de qualidade entre os diferentes modelos. Deve-se levar em conta, por exemplo, que só os micros da linha MSX incluem a acentuação original da língua portuguesa, por terem adotado o padrão BRASCII (ASCII brasileiro).

Outro elemento importante na escolha do hardware para processamento de texto é o vídeo. Nos computadores domésticos mais comuns, este em geral se limita a 32 ou 40 colunas (excepcionalmente 64, como nos micros TRS-80). Esse formato possibilita, entre outras coisas, o uso de televisores como saída de vídeo. Mas é restritivo para um trabalho mais intenso com processadores de texto, principalmente os do tipo WYSWYG, que imprimem uma folha de sessenta a setenta toques.

Em alguns micros, como o MSX e o Apple, é possível ampliar o formato de vídeo com software ou hardware de expansão. Nesse caso, porém, não se recomenda utilizar aparelhos de TV, pois a imagem perde a nitidez. Para quem pretende trabalhar por muito tempo com um processador, a nitidez de texto e a estabilidade no vídeo são fundamentais — e essas qualidades só são obtidas com monitores profissionais (veja o artigo da página 851).

MEMÓRIA

O tamanho da memória disponível também deve ser considerado na seleção de um bom hardware para processamento de texto. A memória RAM não chega a ser um impedimento, dependendo do software escolhido. Alguns processadores só operam com o texto inteiramente contido na memória. Se ele for demasiado grande, é preciso dividi-lo em seções menores, que são gravadas em fita ou vídeo, o que costuma ser bastante cansativo. Uma maior capacidade da RAM ajuda muito, portanto.

Os processadores mais avançados trabalham à base de *paginação*, ou *memória virtual* — isto é, gravam e lêem essas seções do texto em disco, periodicamente. São, contudo, bem mais exigentes em termos de hardware.

A memória auxiliar é outro item importante. Para quem utiliza pouco o processador e lida apenas com textos pequenos (cartas, digamos), um gravador cassete é perfeitamente aceitável para armazenamento de textos. Para uso mais intenso, entretanto, é praticamente indispensável o acionador de disquetes. Além de maior capacidade e velocidade, ele assegura uma taxa de erros baixa.

O emprego de memória virtual e encadeamento de arquivos (seções separadas de um texto, que podem ser reunidas em qualquer ordem) é necessário quando o texto é limitado apenas pela capacidade da memória auxiliar. Porém, quando os textos são muito grandes, essa solução dificulta o trabalho e impede a utilização de funções globais de edição (busca, substituição etc.)

A IMPRESSORA

Em artigos anteriores (páginas 521 e 648), examinamos os diversos tipos de impressora, suas características e as vantagens que cada modelo oferece em relação ao processamento de texto. Por isso, retomaremos aqui apenas os aspectos mais importantes.

A impressora desempenha um papel fundamental na qualidade do texto impresso (o produto final). Para trabalhos que não requerem uma apresentação perfeita, ou para a obtenção de cópias de "rascunho", uma impressora matricial é perfeitamente adequada. Algumas delas podem produzir impressões em *qualidade carta* (dupla densidade de impressão), com uma aparência bastante semelhante à de um texto datilografado. Se queremos conseguir o máximo de qualidade, porém, devemos usar uma impressora do tipo margarida.

Seja qual for o seu objetivo, não se esqueça de verificar se a impressora é compatível com o software de processamento, sobretudo quanto à capacidade para realce de texto, mudança de fontes, acentuação etc.

QUEM PRECISA DE UM PROCESSADOR?

O processamento de textos por computador oferece enormes vantagens para escritores, jornalistas, professores, cientistas ou quaisquer outros profissionais cujas funções incluam a elaboração de documentos, relatórios e cartas. Se você está em uma dessas categorias e ainda não tem um processador, francamente, não sabemos como pôde passar sem ele, até agora!

Se você é estudante, provavelmente seu grau de utilização não é tão intenso, mas o computador ajuda muito na execução de trabalhos escolares.

De qualquer modo, mesmo que você não se enquadre em nenhum dos casos mencionados, um computador é bem vantajoso em relação a uma máquina de escrever, embora o preço não seja muito diferente (se a máquina for elétrica).

DESENHO ARQUITETÔNICO (2)

Desenhe os móveis que irão compor o ambiente a ser planejado. Depois que todas as peças se encaixarem em seus respectivos lugares, restará salvar o projeto em fita ou disco.

Este é o segundo e último artigo sobre o projeto arquitetônico computadorizado. Aqui estão os complementos das listagens para cada um dos micros, acrescidas de instruções detalhadas referentes às três primeiras opções do menu. As outras quatro serão idênticas para as três versões.

S

As opções são as seguintes:
- opção 1, desenha o ambiente;
- opção 2, posiciona a mobília;
- opção 3, cria mobília própria;
- opção 4, grava o projeto;
- opção 5, carrega;
- opção 6, imprime;
- opção 7, sai do programa.

A primeira informação que devemos fornecer ao programa é a maior medida (em metros) do ambiente considerado. Digite os dados e tecle **<ENTER>**. Depois disso, aparecerá um segundo menu contendo as seguintes opções:
<ESPAÇO> desenha uma linha;
<J> desenha uma janela;
<P> desenha uma porta;
<B> movimenta sem desenhar;
<S> volta ao menu principal.

Ao selecionarmos qualquer opção, à exceção da **<S>**, seremos perguntados sobre a direção e o comprimento da reta a ser feita. As questões serão repetidas, de forma a permitir que tracemos linhas na diagonal, por exemplo, dizendo o quanto para cima ou para a direita queremos a reta. Se desejarmos traçá-la simplesmente na horizontal ou na vertical, bastará que forneçamos valores nulos. Repetiremos o processo até concluir o desenho do ambiente e, em seguida, teclaremos **<S>** para retornar ao menu principal. Com um pouco de prática, desenvolveremos nossos projetos em questão de minutos.

Na opção 2, usaremos as teclas **6** e **7** para selecionar o móvel a ser posicionado. Teclando **<S>**, a peça aparecerá no centro da tela. Poderemos movimentá-la para cima, para baixo, para a direita e para a esquerda acionando as teclas **5**, **6**, **7** e **8**. Para girar o móvel no sentido horário, pressionaremos a tecla **H**; no sentido inverso, **A**.

Na criação da mobília, na opção 3, utilizaremos o mesmo processo da opção 1 (desenho do ambiente), com a diferença de que agora o objeto não aparecerá na tela até que sejam fornecidas todas as suas dimensões. Indagados sobre o número de lados da peça (1 a 15) e sobre seu código, daremos um nome de dois dígitos à mobília, de modo a podermos identificá-la mais tarde.

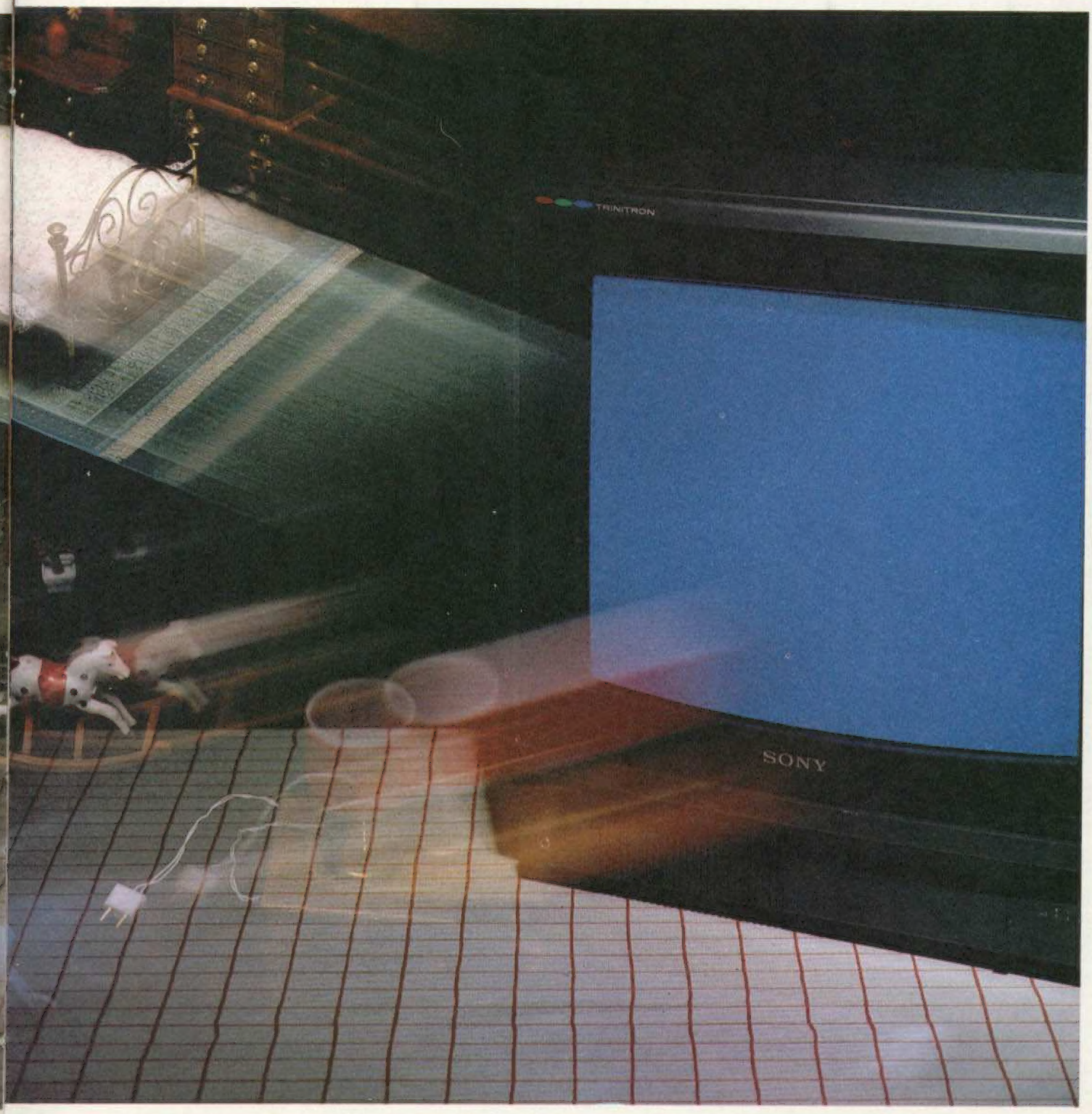
A opção 4 grava tanto a tela como os dados correspondentes às peças da mobília. Usaremos depois a opção 5 para carregar na memória um projeto gravado anteriormente. A opção 6 — de impressão — funcionará somente com a impressora conectada.

```
7010 INPUT "DIGITE DIRECAO  (C,
B,E,D) ?";D$
7020 INPUT "DIGITE DISTANCIA ?"
;D: RETURN
7040 LET Z=CODE INKEY$: FOR K=1
 TO LEN K$: IF CODE (K$(K))=Z T
HEN  LET K=LEN K$: GOTO 7060
7050 NEXT K: GOTO 7040
7060 NEXT K: RETURN
8000 DIM S(10): LET NF=0: LET F
$="000": LET MAX=5: LET R=0: LE
T SC=175/MAX: GOSUB 6060
8010 DIM O(10,30): DIM O$(10,2)
8020 FOR N=1 TO 5: FOR K=1 TO 8
: READ O(N,K): NEXT K: NEXT N
8030 FOR N=1 TO 10: READ O$(N):
 NEXT N
8032 FOR N=USR "A" TO USR "A"+7
: READ A: POKE N,A: NEXT N
8035 FOR N=1 TO 10: READ S(N):
NEXT N
8040 RETURN
8090 DATA .5,0,0,-.5,-.5,0,0,.5
8100 DATA .75,0,0,-.5,-.75,0,0,
.5
8110 DATA .5,0,0,-.75,-.5,0,0,.
75
8120 DATA .5,0,0,-.5,-.5,0,0,.5
8130 DATA .5,0,0,-.5,-.5,0,0,.5
8150 DATA "AR","FO","LL","PI","
GE","XX","XX","XX","XX","XX"
8160 DATA 16,32,64,255,64,32,16
,0
8170 DATA 4,4,4,4,4,0,0,0,0,0
```

T

Ao rodarmos o programa, veremos um menu com as seguintes opções:
- opção 1, desenha o ambiente;
- opção 2, posiciona a mobília;
- opção 3, cria novas peças;
- opção 4, grava o projeto;

- COMPLETE O DESENHO DO AMBIENTE
- CRIE NOVAS PEÇAS PARA SUA MOBÍLIA
- GRAVE O PROJETO
- CARREGUE
- IMPRIMA
- GIRE SEUS MÓVEIS
- APAGUE E REDESENHE
- PROGRAMA TERMINADO



- opção 5, carrega;
- opção 6, imprime;
- opção 7, sai do programa.

Se escolhermos a opção 1, seremos perguntados sobre a maior medida do ambiente. Com essa informação, o micro colocará o desenho em escala, de modo que ele caiba na tela.

Tecle a barra de espaço para iniciar o desenho do ambiente. Há outras opções dentro da opção 1, a saber:
<**ESPAÇO**> para desenhar;
<**B**> para mover sem desenhar;
<**W**> para desenhar janelas;
<**O**> para desenhar portas;
<**C**> para limpar a tela;
<**F**> para voltar ao menu inicial.

Ao escolher uma das opções (à exceção da <**C**> e da <**F**>), o usuário será indagado sobre a direção e o comprimento da reta desejada. Essas perguntas serão feitas várias vezes, a fim de oferecer dados ao computador.

Suponhamos que queremos traçar uma reta inclinada, por exemplo. Basicamente, teremos que fornecer ao micro duas informações, ou seja, o quanto ela deverá se deslocar para cima ou para baixo e o quanto deverá ir para a direita ou a esquerda. A repetição das questões nos permite, portanto, montar coordenadas bidirecionais para a nossa reta e traçá-la na diagonal.

Para obter retas horizontais ou verticais (unidirecionais), bastará teclar <**ENTER**>, quando a pergunta for repetida. Tecle <**C**> para apagar a tela e <**F**> para voltar ao menu principal.

A opção 2 mantém na tela o desenho do ambiente executado na primeira opção e acrescenta um cursor em forma de cruz que será movimentado ponto a ponto através das teclas das setas. Para movê-lo mais rapidamente, tecle <**SHIFT**> + seta, até alcançar a posição desejada — aquela em que o móvel será colocado. Em seguida, acione a tecla **P**. Feito isso, escolha a peça a ser posicionada. As peças são numeradas de 0 a 9; caberá ao usuário criar as cinco últimas, já que as iniciais são automaticamente definidas (mas podem ser alteradas) no programa: 0-pia, 1-fogão, 2-mesa, 3-geladeira e 4-armário. Para apagar um móvel, posicione o cursor no vértice que o gerou, tecle **D** e selecione o número desse móvel. Deveremos nos certificar de que o cursor esteja exatamente no vértice que o gerou; caso contrário, a peça não será totalmente eliminada.

Se quisermos girar um móvel para adequá-lo a um canto qualquer do ambiente, basta acionar a tecla **R** seguido do ângulo de giro do móvel — expresso em graus. Todos os objetos serão desenhados com essa angulação até que se tecle novamente **R**; só que desta vez seguido de 0. A opção < **F** > possibilita retornar ao meu principal.

A opção 3 permite definir novas peças ou modificar as cinco já existentes. As medidas, fornecidas em centímetros, não deverão ultrapassar os 200 cm. O primeiro passo consiste em escolher um

Com o auxílio do computador, pode-se projetar o arranjo de uma cozinha, como mostra a tela à esquerda, em alguns minutos. O menu do programa, exibido na tela abaixo, oferece ao usuário sete opções.

número de 0 a 9 para armazenagem da peça que, em seguida, será desenhada pelo mesmo processo utilizado na opção 1 (desenho do ambiente). Tecle <**C**> para recomeçar o desenho e <**F**> para retornar ao menu inicial.

A opção 4 armazena nosso projeto (que deve receber um nome aqui) em fita ou disco. A opção 5 carrega na memória do micro um projeto previamente gravado em fita ou disco.

A opção de impressão está incompleta, pois necessita de uma rotina que despeje o conteúdo da tela na impressora. Retomaremos este assunto em um próximo artigo, que nos habilitará a elaborar tal rotina.

```
1210 OX=X:OY=Y
1220 IF I$=CHR$(8) AND X>0 THEN
 X=X-1
1230 IF I$=CHR$(9) AND X<185 TH
EN X=X+1
1240 IF I$=CHR$(94) AND Y>0 THE
N Y=Y-1
1250 IF I$=CHR$(10) AND Y<185 T
HEN Y=Y+1
1260 IF I$=CHR$(21) AND X>7 THE
N X=X-8
1270 IF I$=CHR$(93) AND X<177 T
HEN X=X+8
1280 IF I$=CHR$(95) AND Y>7 THE
N Y=Y-8
1290 IF I$=CHR$(91) AND Y<177 T
HEN Y=Y+8
1300 PUT(OX-3,OY-3)-(OX+3,OY+3)
,S,PSET
1310 IF I$="P" THEN 1360
1320 IF I$="D" THEN 1430
1330 IF I$="R" THEN 1500
1340 IF I$="F" THEN FOR K=1 TO
4:PCOPY K TO K+4:NEXT:GOTO 90
1350 GOTO 1160
1360 SOUND 190,1
1370 COLOR 0:GOSUB 340
1380 I$=INKEY$:IF I$<"0" OR I$>
"9" THEN 1380
1390 COLOR 1:GOSUB 340
1400 N=VAL(I$)
1410 COLOR 0,1:A$=O$(N):GOSUB 1
540
1420 GOTO 1160
1430 SOUND 200,1
1440 COLOR 0:GOSUB 340
1450 I$=INKEY$:IF I$<"0" OR I$>
"9" THEN 1450
1460 COLOR 1:GOSUB 340
1470 N=VAL(I$)
1480 COLOR 1,1:A$=O$(N):GOSUB 1
540
1490 GOTO 1160
1500 CLS:PRINT"ANGULO DE ROTACA
O (0-360)":INPUT RT
1510 IF RT<0 OR RT>360 THEN 150
0
1520 RT=(RT/180)*3.141
1530 SCREEN 1,0:GOTO 1160
1540 IF A$="" THEN RETURN
1550 P=1:X1=X:Y1=Y
1560 B$=MID$(A$,P,1)
1570 D$=MID$(A$,P+1,1)
1580 D=VAL(MID$(A$,P+2,INSTR(P,
A$,";")-(P+2)))
1590 P=INSTR(P,A$,";")+1
1600 D=D/100:D2=FNA(D):OX=X:OY=
Y
1610 IF D$="D" THEN XA=D2*COS(R
T):YA=-D2*SIN(RT)
1620 IF D$="E" THEN XA=-D2*COS(
RT):YA=D2*SIN(RT)
1630 IF D$="B" THEN XA=D2*SIN(R
T):YA=D2*COS(RT)
1640 IF D$="C" THEN XA=-D2*SIN(
RT):YA=-D2*COS(RT)
1650 XA=INT(XA+.5):YA=INT(YA+.5
)
1660 X=X+XA:Y=Y+YA
1670 IF B$="D" THEN LINE(OX,OY)
-(X,Y),PSET
1680 IF P<LEN(A$) THEN 1560
1690 X=X1:Y=Y1:RETURN
1700 CLS:LINE INPUT"NOME DO ARQ
UIVO:";F$
1710 SG=PEEK(188)*256
1720 CLS:PRINT"SALVANDO:";F$
1730 CSAVEM F$,SG,SG+6143,35252
1740 GOTO 90
1750 CLS:LINE INPUT "NOME DO AR
QUIVO:";F$
1760 PRINT"ACIONE O GRAVADOR E
ESPERE"
1770 CLOADM F$
1780 GOTO 90
1790 REM **********************
1791 REM *                    *
1792 REM *     ROTINA PARA    *
1793 REM *  DESPEJAR A TELA   *
1794 REM *    NA IMPRESSORA   *
1795 REM *                    *
1796 REM **********************
1797 GOTO 90
```

Ao rodarmos o programa, veremos na tela um menu com sete opções:
- opção 1, desenha o ambiente;
- opção 2, posiciona a mobília;
- opção 3, cria novas peças;
- opção 4, grava o projeto;
- opção 5, carrega;
- opção 6, imprime;
- opção 7, sai do programa.

Devemos primeiro escolher entre as opções 1, 3 ou 5. Ao decidirmos pela 1, seremos perguntados sobre a maior medida (em metros) do ambiente, informação que permitirá ao micro colocar o desenho em uma escala que o faça caber na tela. Tecle a barra de espaço para iniciar o desenho do ambiente.

Há outras opções dentro da opção 1:
<ESPAÇO> para desenhar;
<B> para mover sem desenhar;
<W> para desenhar janelas;
<O> para desenhar portas;
<C> para limpar a tela;
<F> para voltar ao menu inicial.

Ao escolher uma das opções (à exceção da <C> e da <F>), o usuário será perguntado sobre a direção e o comprimento da reta desejada. Essas perguntas serão feitas várias vezes a fim de oferecer dados ao computador.

Suponhamos que queremos traçar uma reta inclinada, por exemplo. Basicamente, teremos que fornecer ao micro duas informações, ou seja, o quanto ela deverá se deslocar para cima ou para baixo e o quanto deverá ir para a direita ou a esquerda. A repetição das questões permite, portanto, montar coordenadas bidirecionais para nossa reta e traçá-la na diagonal.

Para desenhar retas horizontais ou verticais (unidirecionais), basta teclar <ENTER> quando a pergunta for repetida. Tecle <C> para apagar a tela e <F> para voltar ao menu principal.

A opção 2 só poderá ser iniciada depois da opção 1. Ela mantém na tela o desenho do ambiente executado na primeira opção e acrescenta um cursor em forma de seta que será movimentado ponto a ponto através das teclas das setas. Para movê-lo mais rapidamente, usamos as teclas **I**-cima, **M**-baixo, **K**-direita e **J**-esquerda, até que o cursor alcance a posição desejada — aquela em que a peça será colocada. Então acionamos a tecla **P**. Feito isso, determinamos a peça a ser posicionada. As peças são numeradas de 0 a 9; caberá ao usuário criar as cinco últimas peças, já que as iniciais são automaticamente definidas (mas podem ser alteradas) no programa: 0-pia, 1-fogão, 2-mesa, 3-geladeira e 4-armário. Para apagar um móvel, posicione o cursor no vértice que o gerou, tecle **D** e selecione o número correspondente a esse móvel. Deveremos nos certificar de que o cursor esteja exatamente no vértice que o gerou; caso contrário, a peça não será totalmente eliminada.

Se quisermos girar um móvel para fazê-lo caber em algum canto do ambiente, basta acionar **R** seguido do ângulo de giro do móvel, expresso em graus. Todos os objetos serão desenhados com essa angulação, até que se tecle novamente **R**, seguido de 0. A opção <F> permite retornar ao menu principal.

A opção 3 possibilita definir novas peças ou modificar as cinco já existentes. As medidas, fornecidas em centímetros, não deverão ultrapassar os 200 cm. O primeiro passo consiste em escolher um número de 0 a 9 para armazenagem da peça que, em seguida, será desenhada pelo mesmo processo utilizado na opção 1 (desenho do ambiente). Tecle <C> para recomeçar o desenho, e <F> para retornar ao menu inicial.

A opção 4 armazena nosso projeto (que aqui deve receber um nome) em fita. A opção 5 carrega na memória do micro um projeto gravado em fita. Estas duas opções podem ser alteradas para trabalhar com disco, bastando fazer as seguintes modificações: Troque o "**CAS**:" que vem logo após os comandos **BSAVE** e **BLOAD** por "**A**:" (linhas 1720 e 1770), respectivamente.

A opção de impressão está incompleta, pois necessita de uma rotina que despeje o conteúdo da tela na impressora. Retomaremos esse assunto em um próximo artigo, que nos habilitará a elaborar tal rotina.

```
1210 OX=X:OY=Y
1220 IF I$=CHR$(29) AND X>0 THE
N X=X-1
1230 IF I$=CHR$(28)AND X<185THE
N X=X+1
1240 IF I$=CHR$(30) AND Y>0 THE
N Y=Y-1
1250 IF I$=CHR$(31)AND Y<185THE
N Y=Y+1
1260 IF I$=CHR$(74) AND X>7 THE
N X=X-8
1270 IF I$=CHR$(75)AND X<177THE
N X=X+8
1280 IF I$=CHR$(73) AND Y>7 THE
N Y=Y-8
1290 IF I$=CHR$(77)AND Y<177THE
N Y=Y+8
1310 IF I$="P" THEN CL=1:GOTO 1
360
1320 IF I$="D" THEN CL=15:GOTO1
430
1330 IF I$="R" THEN A=USR(0):GO
TO 1500
1340 IF I$="F" THEN A=USR(0):GO
TO 90
1350 GOTO 1160
1360 BEEP
1370 GOSUB 340
1380 I$=INKEY$:IF I$<"0" OR I$>
"9" THEN 1380
1390 COLOR 0:GOSUB 340
1400 COLOR 1:N=VAL(I$)
1410 A$=O$(N):GOSUB 1540
1420 GOTO 1160
1430 BEEP
1440 GOSUB 340
1450 I$=INKEY$:IF I$<"0" OR I$>
"9" THEN 1450
1460 COLOR 0:GOSUB 340
1470 N=VAL(I$)
1480 A$=O$(N):GOSUB 1540
1490 GOTO 1160
1500 SCREEN0:PRINT"ANGULO DE RO
TAÇÃO (0-360)";:INPUT RT
1510 IF RT<0 OR RT>360 THEN 150
0
1520 RT=(RT/180)*3.141:SCREEN 2
1530 A=USR2(0):A=USR1(0):GOTO 1
160
1540 IF A$="" THEN RETURN
1550 P=1:X1=X:Y1=Y
1560 B$=MID$(A$,P,1)
1570 D$=MID$(A$,P+1,1)
1580 D=VAL(MID$(A$,P+2,INSTR(P,
A$,";")-(P+2)))
1590 P=INSTR(P,A$,";")+1
1600 D=D/100:D2=FNA(D):OX=X:OY=
Y
1610 IF D$="D" THEN XA=D2*COS(R
T):YA=-D2*SIN(RT)
1620 IF D$="E" THEN XA=-D2*COS(
RT):YA=D2*SIN(RT)
1630 IF D$="B" THEN XA=D2*SIN(R
T):YA=D2*COS(RT)
1640 IF D$="C" THEN XA=-D2*SIN(
RT):YA=-D2*COS(RT)
1650 XA=INT(XA+.5):YA=INT(YA+.5
)
1660 X=X+XA:Y=Y+YA
1670 IFB$="D" THENLINE(OX,OY)-(
X,Y),CL
1675 COLOR 1
1680 IF P<LEN(A$) THEN 1560
1690 X=X1:Y=Y1:RETURN
1700 LOCATE 5,18:LINE INPUT"NOM
E DO PROJETO:";F$
1710 LOCATE 5,20:PRINT"GRAVANDO
 "+F$+"..."
1720 BSAVE "CAS:"+F$,&HD100,&HE
900
1730 GOTO 90
1750 LOCATE 5,20:LINE INPUT"NOM
E DO PROJETO:";F$
1760 LOCATE 5,22:PRINT"CARREGAN
DO "+F$+"..."
1770 BLOAD "CAS:"+F$
1780 F1=1:GOTO 90
1790 REM **********************
1791 REM *                    *
1792 REM *   rotina para      *
1793 REM * despejar a tela    *
1794 REM *  na impressora     *
1795 REM *                    *
1796 REM **********************
1797 GOTO 90
2000 CLEAR 200,&HA000
2010 DEFUSR=&HA000
2015 DEFUSR1=&HA00D
2020 DEFUSR2=&HA01A
2025 DEFUSR3=&HA026
2030 DEFUSR4=&HA033
2040 FOR R=0 TO 63:READ A
2050 POKE(&HA000+R),A:NEXT
2060 DATA &H21,00,00,&H11,00,&H
D1,01,00,24,&HCD,&H59,00,&HC9
2070 DATA &H21,00,&HD1,&H11,00,
00,01,00,24,&HCD,&H5C,00,&HC9
2080 DATA &H21,00,&H20,62,00,01
,00,24,&HCD,&H56,00,&HC9
2090 DATA &H21,00,00,&H11,&HFF,
&HB8,01,00,24,&HCD,&H59,00,&HC9
2100 DATA &H21,&HFF,&HB8,&H11,0
0,00,01,00,24,&HCD,&H5C,00,&HC9
2110 POKE(&HA01E),31
2120 SP$="":FORI=1TO8:READA$
2130 SP$=SP$+CHR$(VAL("&H"+A$))
2140 NEXT:SPRITE$(0)=SP$
2150 GOTO 20
2160 DATA E0,C0,A0,90,08,00,00,
00
```

DESENHOS EM PERSPECTIVA

- PERSPECTIVA
- PONTO DE FUGA E PONTO DE VISTA
- DIMINUIÇÃO DE TAMANHO
- TÉCNICA DE SOMBREAMENTO

Traga seu desenho para a magia do mundo tridimensional. Isso é possível graças aos recursos de seu micro aliados aos princípios da perspectiva e das técnicas de sombreamento.

A maioria das pessoas sabe da importância do uso da perspectiva em qualquer figura, e, por isso, costuma incluí-la até nos mais simples esboços. Surpreendentemente, a idéia de perspectiva não é tão clara como parece; assim, os artistas mais antigos não tinham a menor idéia sobre o que fazer para dar profundidade às suas telas. Apenas a partir do Renascimento, quando os pintores começaram a dar um tom realista às representações, é que os princípios de perspectiva foram formulados. Essas regras são as mesmas usadas pelos artistas contemporâneos e aplicadas aos desenhos feitos no computador.

Para o desenvolvimento dessa técnica, foram necessários vários séculos, o que não significa que suas noções básicas sejam de difícil compreensão. Na realidade, a idéia fundamental é a seguinte: todas as linhas horizontais permanecem nessa posição, enquanto as verticais se inclinam "para dentro" da figura, convergindo para o denominado *ponto de fuga*. (Esse assunto já foi abordado no artigo *Programação de gráficos em 3-D*, página 693.)

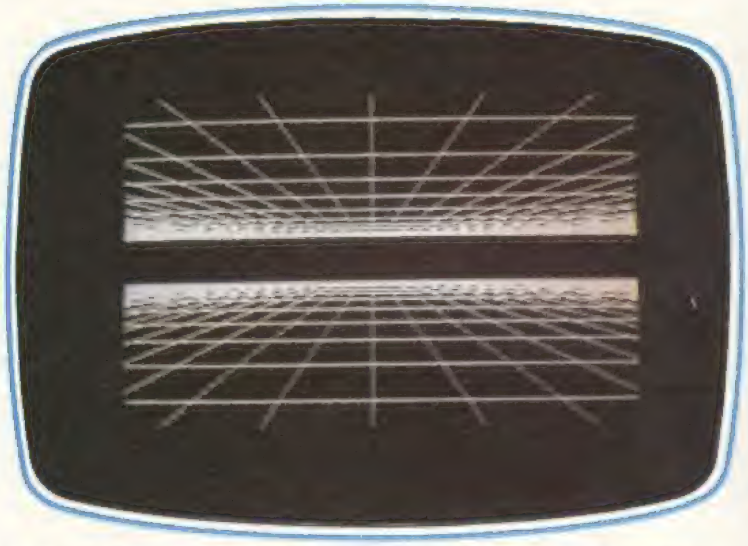
Apenas as linhas paralelas, perpendiculares à figura, se dirigem para o ponto de fuga principal. As demais afluem para seus próprios pontos de fuga, situados em uma mesma linha, no horizonte imaginário do desenho.

Esses princípios, quando aplicados a figuras reais, podem se tornar um pouco mais complicados; em desenhos simples, entretanto, proporcionam a sensação de profundidade exigida. O leitor já deve ter visto, na televisão ou no cinema, astronaves voando para o espaço a partir do imenso hangar da nave-mãe. Nesse caso, a sensação de profundidade é dada pelo chão e pelo teto quadriculados do hangar, que parecem se projetar em direção ao espaço.

FEIXES DE RETA EM PERSPECTIVA

Os feixes de retas perpendiculares são muito usados para se obter a visão de

um objeto em perspectiva. O primeiro programa, mostrado a seguir, desenha duas superfícies quadriculadas — o teto e o chão do hangar — tendo o usuário do micro como observador, olhando para o infinito.



Um cenário quadriculado em perspectiva é um bom recurso para conferir profundidade a seu desenho.

```
 10 BORDER 0: PAPER 0: INK 7
 20 CLS : INPUT "PROXIMIDADE D
O CHAO ";V
 40 FOR K=-126 TO 127 STEP 6
 50 LET Y=174: LET X=K+V*K: IF
ABS (X)>127+(X<1) THEN  LET X
=SGN (X)*127-(X<1): LET Y=100
+(X-K)*74/(V*K)
 60 PLOT 127-K,100: DRAW 127-X
-PEEK 23677,Y-PEEK 23678
 70 PLOT 127-K,74: DRAW 127-X-
PEEK 23677,174-Y-PEEK 23678
 80 NEXT K
 90 LET F=V^(1/6): LET Y=F
100 LET Y=Y*F: IF Y>77 THEN
GOTO 140
110 PLOT 0,97+Y: DRAW 255,0
120 PLOT 0,77-Y: DRAW 255,0
130 GOTO 100
140 IF INKEY$="" THEN  GOTO
 140
150 GOTO 20
```

```
10 PMODE 4,1
20 CLS:INPUT"PROXIMIDADE DO CHA
O ";V
30 PCLS:SCREEN 1,1
40 FOR K=-126 TO 127 STEP 6
50 Y=191:X=K+V*K:IF ABS(X)>127-
(X<1) THEN X=SGN(X)*127+(X<1):Y
=111+(X-K)*80/(V*K)
60 LINE(127-K,111)-(127-X,Y),PS
ET
70 LINE(127-K,80)-(127-X,191-Y)
,PSET
80 NEXT
90 F=V↑(1/6):Y=F
100 Y=Y*F:IF Y>83 THEN 140
110 LINE(0,108+Y)-(255,108+Y),P
SET
120 LINE(0,84-Y)-(255,84-Y),PSE
T
130 GOTO 100
140 IF INKEY$="" THEN 140 ELSE
20
```

```
10  TEXT
20  HOME : INPUT "PROXIMIDADE D
O CHAO ";V
30  HGR2
40  FOR K =  - 126 TO 127 STEP
6
50 Y = 191:X = K + V * K: IF  A
BS (X) > 127 - (X < 1) THEN X =
  SGN (X) * 127 + (X < 1):Y = 1
11 + (X - K) * 80 / (V * K)
60  HPLOT 127 - K,111 TO 127 -
X,Y
70  HPLOT 127 - K,80 TO 127 - X
,191 - Y
80  NEXT
90 F = V ^ (1 / 6):Y = F
100 Y = Y * F: IF Y > 83 THEN 1
40
110  HPLOT 0,108 + Y TO 255,108
 + Y
120  HPLOT 0,84 - Y TO 255,84 -
 Y
130  GOTO 100
140  GET A$: GOTO 10
```

```
10 SCREEN 0
20 CLS:INPUT"PROXIMIDADE DO CHA
O ";V
30 SCREEN 2
40 FOR K=-126 TO 127 STEP 6
50 Y=191:X=K+V*K:IF ABS(X)>127-
(X<1) THEN X=SGN(X)*127+(X<1):Y
=111+(X-K)*80/(V*K)
60 LINE(127-K,111)-(127-X,Y)
70 LINE(127-K,80)-(127-X,191-Y)
80 NEXT
90 F=V^(1/6):Y=F
100 Y=Y*F:IF Y>83 THEN 140
110 LINE(0,108+Y)-(255,108+Y)
120 LINE(0,84-Y)-(255,84-Y)
130 GOTO 100
140 IF INKEY$="" THEN 140 ELSE
20
```

Alterando o valor inicial, obtêm-se efeitos diferentes, como a sensação de estar flutuando entre as duas superfícies. Atribuindo um valor baixo para **V** (2, por exemplo), as linhas inferiores se inclinarão para os pontos de fuga situados em um horizonte acima da metade do vídeo. Inversamente, as linhas superiores convergirão para um horizonte abaixo da metade do vídeo. Se, por outro lado, digitarmos o valor 10, o cenário apresentará modificações profundas, com o observador em uma posição bem mais próxima do chão.

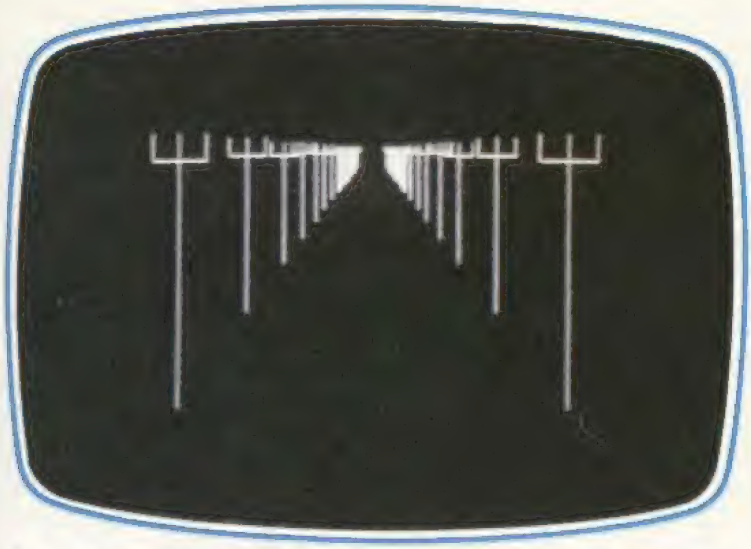
Inicialmente, o programa desenha as retas frontais voltadas para o horizonte, entre as linhas 40 e 80, e as retas que vão de lado a lado da tela, entre as linhas 90 e 130. A variável **K** dá a coordenada X do início de cada linha, que é, então, multiplicada por **V** para chegar à coordenada X da sua extremidade final. A condição **IF...THEN**, na linha 50, apenas evita que o desenho ultrapasse os limites da tela. Em alguns computadores, isto é dispensável, apesar de acelerar a execução do programa.

As linhas restantes serão traçadas pela próxima parte do programa. A distância entre elas torna-se cada vez menor à medida que "se afastam" da tela, um efeito controlado por sucessivas multiplicações da coordenada Y por **V↑1/6**. O valor 1/6 foi escolhido para dar um resultado mais realista; isto, porém, não impede que o usuário experimente outros valores.

A DIMINUIÇÃO DE TAMANHO

Os princípios da perspectiva aplicam-se aos objetos bem como aos feixes de retas. Vistos a distância, todos os elementos de um desenho parecem menores e mais próximos uns dos outros. Isso se verifica, por exemplo, quando observamos os topos e as bases de uma série de árvores iguais enfileiradas colocadas sobre duas retas que convergem para um mesmo ponto de fuga.

Obtém-se esse efeito porque o céré-



À medida que nos afastamos dos objetos, eles parecem menores e mais próximos entre si, como mostra a figura acima.

São necessários princípios e técnicas especiais para se obter um bom sombreamento em esferas.

bro humano relaciona o tamanho de um objeto conhecido com a distância em que ele se encontra. Ao vermos dois objetos idênticos na forma, mas um com a metade do tamanho do outro, deduziremos que o menor está duas vezes mais longe que o primeiro. Portanto, ao desenharmos alguns objetos em tamanho decrescente no vídeo de um micro, teremos a impressão de que eles estarão "entrando na tela".

Analisando a distância entre os objetos de uma figura, concluiremos que ela se comporta do mesmo modo. Os objetos parecem mais próximos entre si à medida que se afastam do observador.

Existe uma relação matemática entre a distância e o tamanho, isto é, o tamanho apresenta uma variação *inversamente* proporcional à distância. Assim, se aumentarmos a distância entre o objeto e o observador, o tamanho do objeto diminuirá. É por essa razão que no primeiro programa a separação entre as linhas horizontais está elevada a 1/6. O número 6 foi escolhido apenas para se garantir um espaçamento mais condizente com a realidade; se o substituirmos por qualquer outro, veremos que o desenho não perderá sua profundidade.

O próximo programa mostra como desenhar uma vista em perspectiva de uma estrada delimitada por postes:

```
10 BORDER 0: PAPER 0: INK 7:
CLS
15 DEF FN Y(X)=((174-VP)/100)
*(X-128)+VP
16 DEF FN B(X)=((20-VP)/100)*
(X-128)+VP
17 DEF FN S(T)=SF/SQR ((RW/2)
^2+(T*PH)^2)
20 PRINT '''
30 INPUT "DISTANCIA ENTRE POS
TES ";P
40 INPUT "LARGURA DA ESTRADA
";RW
50 INPUT "ALTURA DOS POSTES "
;PH
60 INPUT "ALTURA DO OBSERVADO
R ";RH
70 CLS
80 LET SF=1: LET SF=160/FN S(
0)
90 LET VP=160/PH*RH+100: LET
X=228: FOR T=1 TO 15
100 LET X=X-1: IF FN S(T)<FN Y
(X)-FN B(X) THEN  GOTO 100
110 PLOT X,FN B(X): DRAW X-
PEEK 23677,FN Y(X)-PEEK 23678:
LET XJ=(FN Y(X)-FN B(X))/10:
LET YJ=FN Y(X)-XJ
120 PLOT X-XJ,FN Y(X): DRAW X-
XJ-PEEK 23677,YJ-PEEK 23678:
DRAW X+XJ-PEEK 23677,YJ-PEEK
23678: DRAW X+XJ-PEEK 23677,FN
Y(X)-PEEK 23678
130 PLOT 255-X,FN B(X): DRAW
255-X-PEEK 23677,FN Y(X)-PEEK
23678
140 PLOT 255-X-XJ,FN Y(X):
DRAW 255-X-XJ-PEEK 23677,YJ-
PEEK 23678: DRAW 255-X+XJ-PEEK
23677,YJ-PEEK 23678: DRAW 255-
X+XJ-PEEK 23677,FN Y(X)-PEEK
23678
150 NEXT T
160 GOTO 160
```

```
10 PMODE 4,1:PCLS:CLS
20 DEFFNYT(X)=((900-VP)/500)*(X
-640)+VP:DEFFNYB(X)=((100-VP)/5
00)*(X-640)+VP:DEFFNS(T)=SF/SQR
((RW/2)↑2+(T*PH)↑2)
30 INPUT"DISTANCIA ENTRE OS POS
TES";P
40 INPUT"LARGURA DA ESTRADA";RW
50 INPUT"ALTURA DOS POSTES";PH
60 INPUT"ALTURA DO OBSERVADOR";
RH
70 SCREEN 1,1
80 SF=1:SF=800/FNS(0)
90 VP=800/PH*RH+100:X=1140:FOR
T=1 TO 15
100 X=X-4:IF FNS(T)<FNYT(X)-FNY
B(X) THEN 100
110 LINE(X/5,191-FNYB(X)/5)-(X/
5,191-FNYT(X)/5),PSET:XJ=(FNYT(
X)-FNYB(X))/10:YJ=FNYT(X)-XJ
120 LINE((X-XJ)/5,191-FNYT(X)/5
)-((X-XJ)/5,191-YJ/5),PSET:LINE
-((X+XJ)/5,191-YJ/5),PSET:LINE-
((X+XJ)/5,191-FNYT(X)/5),PSET
130 LINE(255-X/5,191-FNYB(X)/5)
-(255-X/5,191-FNYT(X)/5),PSET
140 LINE(255-(X+XJ)/5,191-FNYT(
X)/5)-(255-(X+XJ)/5,191-YJ/5),P
SET:LINE -(255-(X-XJ)/5,191-YJ/
5),PSET:LINE-(255-(X-XJ)/5,191-
YJ/5),PSET:LINE-(255-(X-XJ)/5,1
91-FNYT(X)/5),PSET
150 NEXT
160 IF INKEY$="" THEN 160 ELSE
RUN
```

```
10  TEXT : HOME
20  DEF  FN YT(X) = ((900 - VP)
 / 500) * (X - 640) + VP: DEF
FN YB(X) = ((100 - VP) / 500) *
 (X - 640) + VP: DEF  FN S(T) =
 SF /  SQR ((RW / 2) ^ 2 + (T *
 PH) ^ 2)
30  INPUT "DISTANCIA ENTRE OS P
OSTES ";P
40  INPUT "LARGURA DA ESTRADA "
;RW
50  INPUT "ALTURA DOS POSTES ";
PH
60  INPUT "ALTURA DO OBSERVADOR
 ";RH
70  HCOLOR= 3: HGR2
```

```
80 SF = 1:SF = 800 /  FN S(0)
90 VP = 800 / PH * RH + 100:X =
 1140: FOR T = 1 TO 15
100 X = X - 4: IF  FN S(T) <  F
N YT(X) -  FN YB(X) THEN 100
110  HPLOT X / 5,191 -  FN YB(X
) / 5 TO X / 5,191 -  FN YT(X)
/ 5:XJ = ( FN YT(X) -  FN YB(X)
) / 10:YJ =  FN YT(X) - XJ
120  HPLOT (X - XJ) / 5,191 -
FN YT(X) / 5 TO (X - XJ) / 5,19
1 - YJ / 5: HPLOT  TO (X + XJ)
/ 5,191 - YJ / 5: HPLOT  TO (X
+ XJ) / 5,191 -  FN YT(X) / 5
130  HPLOT 255 - X / 5,191 -  F
N YB(X) / 5 TO 255 - X / 5,191
-  FN YT(X) / 5
140  HPLOT 255 - (X + XJ) / 5,1
91 -  FN YT(X) / 5 TO 255 - (X
+ XJ) / 5,191 - YJ / 5: HPLOT
TO 255 - (X - XJ) / 5,191 - YJ
/ 5: HPLOT  TO 255 - (X - XJ) /
 5,191 -  FN YT(X) / 5
150  NEXT
160  GET A$: RUN
```

```
10 SCREEN 0
20 DEFFNYT(X)=((900-VP)/500)*(X
-640)+VP:DEFFNYB(X)=((100-VP)/5
00)*(X-640)+VP:DEFFNS(T)=SF/SQR
((RW/2)^2+(T*PH)^2)
30 INPUT"DISTANCIA ENTRE OS POS
TES ";P
40 INPUT"LARGURA DA ESTRADA ";R
W
50 INPUT"ALTURA DOS POSTES ";PH
60 INPUT"ALTURA DO OBSERVADOR "
;RH
70 SCREEN 2
80 SF=1:SF=800/FNS(0)
90 VP=800/PH*RH+100:X=1140:FOR
T=1 TO 15
100 X=X-4:IF FNS(T)<FNYT(X)-FNY
B(X) THEN 100
110 LINE (X/5,191-FNYB(X)/5)-(X
/5,191-FNYT(X)/5):XJ=(FNYT(X)-F
NYB(X))/10:YJ=FNYT(X)-XJ
120 LINE ((X-XJ)/5,191-FNYT(X)/
5)-((X-XJ)/5,191-YJ/5):LINE-((X
+XJ)/5,191-YJ/5):LINE-((X+XJ)/5
,191-FNYT(X)/5)
130 LINE (255-X/5,191-FNYB(X)/5
)-(255-X/5,191-FNYT(X)/5)
140 LINE (255-(X+XJ)/5,191-FNYT
(X)/5)-(255-(X+XJ)/5,191-YJ/5):
LINE-(255-(X-XJ)/5,191-YJ/5):LI
NE-(255-(X-XJ)/5,191-FNYT(X)/5)
150 NEXT
160 IF INKEY$="" THEN 160 ELSE
RUN
```

Este programa permite ao usuário definir exatamente o cenário da estrada antes de desenhá-lo em perspectiva.

Para facilitar os cálculos, ele oferece três funções: **FNS(T)** calcula a altura de cada poste; **FNYT(X)** e **FNYB(X)** dão as coordenadas Y da extremidade inferior e superior dos postes para qualquer posição X. Podemos imaginar o trabalho dessas duas funções como sendo o de traçar duas retas que afluem para um determinado ponto de fuga, entre as quais serão colocados os postes.

Na parte principal do programa, a linha 80 calcula o fator de escala, **SF**, que constrói um poste tomando por base a altura do anteriormente desenhado. A variável **VP** define o valor da coordenada Y do ponto de fuga, que, por sua vez, depende da altura do observador e da altura dos postes. A coordenada X está sempre no centro da tela.

Quinze postes são desenhados em cada lado da estrada pelo laço entre as linhas 90 e 150. Os da margem direita são os primeiros a serem erguidos.

Inicialmente, a linha 100, começando a partir da extremidade direita da tela, decresce o valor da posição X, até que a altura do poste, calculada por **FNS(T)**, se encaixe exatamente entre as linhas criadas por **FNYT** e **FNYB**. Feito isso, a primeira parte da linha 110 desenha o poste, enquanto a porção restante manipula as variáveis **XJ** e **YJ**, que são fatores de escala para a construção da parte superior (suporte dos fios). A linha 120 é responsável por essa última tarefa.

SOMBREAMENTO

Além da perspectiva, existem outras técnicas capazes de dar tridimensionalidade aos gráficos. A mais comum delas é o sombreamento. Infelizmente, a falta de variação de tons das cores constitui um grande problema quando tentamos aplicar essa técnica aos computadores domésticos. Com isso, torna-se extremamente difícil criar o sombreamento necessário para um desenho em três dimensões.

Já os computadores maiores, oferecendo uma rica gama de tonalidades, fazem dessa tarefa algo bem mais fácil.

PLANETAS PONTILHADOS

É possível acrescentar sombras a uma figura qualquer colorindo apenas alguns pontos em sua região mais escura e a maioria deles nas partes mais claras. Portanto, o número de pontos acesos em uma certa área do objeto é que irá determinar a claridade.

Com a variação gradual da densidade nas diferentes regiões, poderemos criar o efeito de sombreamento. O próximo programa utiliza essa técnica para desenhar um grupo de esferas. Elas se parecem tanto com planetas girando no espaço que o programa chega a concluir um fundo estrelado e a desenhar um anel ao redor de algumas delas.

```
10 BRIGHT 0: BORDER 0: PAPER
0: INK 7: CLS
20 FOR I=1 TO 100: PLOT RND*
255,RND*175: NEXT I
25 LET F=0: FOR T=1 TO 3
30 LET CL=RND*5+2
40 LET XC=RND*195+30: LET YC=
RND*115+30
50 LET S=RND*30
60 FOR K=-S TO S
70 IF INT (K)=0 AND F=0 THEN
PLOT  INVERSE 1; OVER 1;XC-L*
2,YC: DRAW  INK CL;L*4,0: LET
F=1
80 LET X=SQR (S*S-K*K)
90 LET X2=2*X
100 FOR L=-X TO X
110 PLOT  INK CL; INVERSE 1;XC
+L,YC+K: IF RND*X2-X<L THEN
PLOT  INK CL;XC+L,YC+K
120 NEXT L
130 NEXT K
140 NEXT T
```

```
150 PAUSE 0
```

```
10 PMODE 3,1:PCLS3:SCREEN 1,0
20 FOR K=1 TO 100:PSET(RND(256)
-1,RND(192)-1,2):NEXT
25 FOR T=1 TO 8
30 XC=RND(195)+30:YC=RND(131)+3
0
40 CL=RND(3):CL=CL-(CL=3)
50 S=RND(25)+5
60 FOR K=-S TO S
70 IF K=1 AND RND(4)=1 THEN CIR
CLE(XC,YC),S*1.5,RND(4),0
80 X=SQR(S*S-K*K)
90 X2=2*X
100 FOR L=-X TO X STEP 2
110 IF RND(X2)-X<L THEN COLOR C
L ELSE COLOR 3
120 PSET(XC+L,YC-K)
130 NEXT L,K
140 NEXT T
150 GOTO 150
```

```
10  HCOLOR= 3: HGR2
20  FOR K = 1 TO 100: HPLOT  IN
T (279 *  RND (1) + 1), INT (19
1 *  RND (1) + 1): NEXT K
25  FOR T = 1 TO 8
30 XC =  INT ( RND (1) * 195 +
30):YC =  INT ( RND (1) * 131 +
 30)
40 CL =  INT ( RND (0) * 3 + 1)
:CL = CL - (CL = 3)
50 S =  INT ( RND (1) * 25 + 5)

60  FOR K =  - S TO S
70  IF K = 1 AND  INT ( RND (1)
 * 4 + 1) = 1 THEN  GOSUB 160
80 X =  SQR (S * S - K * K)
90 X2 = 2 * X
100  FOR L =  - X TO X STEP 2
110  IF  INT ( RND (0) * X2 + 1
) - X < L THEN  HCOLOR= CL: GOT
O 120
115  HCOLOR= 3
120  HPLOT XC + L,YC - K
130  NEXT L,K
140  NEXT T
150  GOTO 150
160  FOR A = 0 TO 6.28 STEP  05

170  HPLOT S * 2 *  COS (A) + X
C,S *  5 *  SIN (A) + YC
180  NEXT A
190  RETURN
```

```
10 COLOR 15,1,1:SCREEN 2
20 FOR K=1 TO 100:PSET(INT(RND(
1)*256),INT(RND(1)*192)):NEXT
25 FOR T=1 TO 8
30 XC=INT(RND(-TIME)*195)+30:YC
=INT(RND(-TIME)*131)+30
40 CL=INT(RND(1)*3):CL=CL-(CL=3
)
50 S=INT(RND(1)*25)+5
60 FOR K=-S TO S
70 IF K=1 AND INT(RND(1)*4)=1 T
HEN CIRCLE(XC,YC),S*1.5,9,0,,.2
5
80 X=SQR(S*S-K*K)
90 X2=2*X
100 FOR L=-X TO X STEP 2
110 IF INT(RND(1)*X2)-X<L THEN
COLOR CL ELSE COLOR 3
120 PSET(XC-L,YC-K)
130 NEXT L,K
140 NEXT T
150 GOTO 150
```

O cenário de estrelas é feito pela linha 20, que acende aleatoriamente 100 pontos na tela. Oito esferas serão desenhadas (nos micros da linha Spectrum apenas três) pelo laço entre as linhas 25 e 140. A linha 30 escolhe uma posição casual para o centro de cada esfera, e as linhas seguintes determinam cor e tamanho.

O procedimento geral para a criação de cada esfera consiste em preenchê-la, de baixo para cima, com retas horizontais. Esse trabalho é controlado por meio da variável **K**, que tem −S como valor inicial (a coordenada Y da parte mais baixa da esfera) e +S como valor final (a coordenada Y da parte mais alta da esfera).

A linha 80 define a coordenada X do início de cada reta através da equação para construção do círculo. **X2** é o comprimento de cada reta.

As linhas 100 e 130 determinam a cor das linhas. A variável **L** pode ser considerada como a luminosidade de cada região; seu valor varia segundo a distância entre o ponto e a borda do círculo. Um número aleatório (dependendo do comprimento da reta) é escolhido pela linha 110; caso seja menor que a claridade da região, define-se o ponto a ser plotado como sendo de cor preta. Se, ao contrário, o número for maior, o ponto será definido como sendo da cor escolhida na linha 40.

Por fim, a linha 135 desenha um anel ao redor de dois ou três planetas.

Experimente adaptar estas rotinas para sombrear outros sólidos, como cilindros ou superfícies planas.

FORMATAÇÃO DE TELAS

Aprenda a montar telas para entrada de dados formatados nos computadores Apple e TK-2000. A padronização tornará seu trabalho mais profissional e confiável.

Se você já se deu ao trabalho de escrever um programa, certamente enfrentou dificuldades na padronização da entrada dos dados. Quantas vezes não foi preciso verificar o número máximo de caracteres de uma cadeia *string*, para que não excedesse o tamanho do seu arquivo, ou manipular uma data para dar-lhe um formato adequado?

A rotina que publicamos aqui resolve esses e outros problemas, padronizando o formato de entrada de dados e, ainda, melhorando sua apresentação. Ela fornece formatos definidos para a entrada de dados numéricos, alfanuméricos, data e hora. Além disso, para que você tenha uma preocupação a menos, encarrega-se de verificar a validade das datas e horas.

COMO USAR A ROTINA

A utilização da rotina de formatação é simples. Digite o programa e execute-o, para ter uma amostra do que ela faz. Antes de chamar a rotina, você precisará apenas entrar um parâmetro que definirá o formato e o tipo de dado que a máquina aceitará.

O parâmetro será fornecido pela variável **P$**, já que o BASIC não permite a passagem direta de parâmetros para as sub-rotinas. Assim, se você quiser dar entrada a uma data, a variável **P$** deverá conter a letra C (faça **P$** = "C" e use "calendário" como mnemônico); para a entrada de hora, **P$** conterá o valor H (**P$** = "H"). Como esses dois formatos são fixos, basta uma só letra.

Para a entrada de números e de variáveis alfanuméricas, mais informações serão necessárias. Na entrada de números, a variável **P$** deverá conter a letra D seguida de dois valores, que indicam o número de algarismos (incluindo o ponto decimal) e o número de casas decimais. O primeiro valor deve ter sempre dois dígitos (até 9, coloque 0 antes do algarismo significativo); o segundo, apenas um dígito (use uma vírgula para separá-lo do anterior). Se você quiser, por exemplo, dar entrada a um número de nove dígitos, dois dos quais decimais, use: **P$** = "**D09,2**". Para a entrada de variáveis alfanuméricas, utiliza-se a letra G (lembre-se de "geral", já que aqui são aceitos números e letras), seguida do número máximo de caracteres a serem digitados. Para a entrada de um nome de até trinta caracteres, use: **P$** = "**G,30**".

Definido o parâmetro, chama-se a sub-rotina por meio do comando **GOSUB 600**. Se for conveniente, você poderá renumerá-la e colocá-la em qualquer parte do seu programa. Nesse caso, não se esqueça de mudar o número da linha pedida pelo **GOSUB**.

Depois de digitada, a informação é armazenada na variável **W$**. Para usar seu conteúdo, é necessário transferi-lo para outra variável cujo valor não seja alterado pela sub-rotina. Uma linha de programa que utiliza a rotina de entrada de dados fica assim:

```
100 PRINT "VALOR DO SALARIO";:
PS="D08,2":GOSUB 600:SA=VAL(WS)
```

Quanto à localização do formato para a entrada de dados na tela, a rotina também é simples. Seu ponto inicial será aquele em que você tiver deixado o cursor antes de chamar a rotina. Para marcar esse ponto, coloque um ponto e vírgula no fim da mensagem. Se essa posição não lhe convier, posicione o cursor com **HTAB** e **VTAB**.

Como está, a rotina não trabalha bem se o formato ficar dividido em duas linhas, pois só o posicionamento horizontal é monitorado. Mais à frente, veremos como modificar essa situação.

FUNCIONAMENTO

A rotina começa na linha 600, onde a variável **PH** passa a conter o valor da atual posição horizontal do cursor. A posição de memória 36 armazena valores de 0 a 39. Somamos 1 ao valor em questão, tornando-o equivalente ao usado por **HTAB**. A variável **PP** passa então a conter o valor ASCII da letra que caracteriza o parâmetro. Com base nessa variável, o programa será desviado adequadamente na linha 604. As linhas indicadas em 604 atribuem os valores necessários às variáveis que controlam o funcionamento da sub-rotina.

Examinemos essas variáveis. Inicialmente, **P** contém o comprimento do dado a ser digitado. Para data e hora, esse valor é fixo; para outros tipos de dado, é lido de **P$**.

PF indica o ponto em que se poderá deixar a rotina. Assim, para entradas alfanuméricas, seu conteúdo é 1 — ou seja, pelo menos um caractere tem que ser digitado, não se aceitando entradas vazias. Nos demais casos, **PF** é igual a **P**, forçando o uso de todos os campos do formato de entrada. Isso impede que uma data seja digitada pela metade ou que se omita a parte decimal de um número (mesmo que seja 0).

Duas outras variáveis, **P1** e **P2**, marcam a posição onde se usam caracteres fixos no formato. Assim, para datas, o caractere "/" é utilizado duas vezes. A posição é contada a partir de 0.

Finalmente, as variáveis alfanuméricas iniciadas por W servem para montar a linha de formatação, que ficará armazenada em **WD$**.

Depois que o conteúdo dessas variáveis foi definido, a execução converge para a linha 618. Aí terá início a entrada de dados propriamente dita. O formato é colocado na posição lida pela linha 600 e as variáveis **W$**, **WI$** e **PX** são limpas, passando a conter a posição do caractere dentro do *string* que está sendo trabalhado.

A linha 626 indica o caractere correspondente à posição atual do cursor, usando operações lógicas. Para isso, o computador atribui valor 1 às relações verdadeiras, e 0, às falsas. Assim, todas as comparações são condensadas em uma única linha, evitando-se o uso de várias instruções **IF**.

O caractere indicado é colocado em sua posição e, em seguida, o cursor é reposicionado. Espera-se então a entrada de um caractere pelo teclado (linha 628). A rotina só particulariza dois tipos de caractere: o *carriage return* (ou seja, <**CR**>, cujo código é 13) e o *backspace* (ou ←, de código 8). Os demais são tratados em grupo.

Ao se digitar <**CR**>, o caractere será reconhecido pelo seu código, que está armazenado na variável **PA**. Note que ele só terá efeito se a posição do caractere atual (**PX**) for maior ou igual ao

- UMA ROTINA PARA ENTRADA DE DADOS
- DEFINIÇÃO DO PARÂMETRO TIPOS DE DADOS
- ARMAZENAGEM DA INFORMAÇÃO
- POSICIONAMENTO DO CURSOR
- COMO FUNCIONA
- APERFEIÇOAMENTOS

ponto de saída definido (**PF**). Assim, se estivermos entrando uma data, não poderemos deixar a rotina antes de digitar o último caractere.

O back-space (linha 632) só terá efeito se estiver além do primeiro caractere (**PX > 0**). Nesse caso, a rotina é desviada para a linha 644 e o valor 1 é subtraído de **PH** — ou seja, a tabulação horizontal retrocede uma posição. Se **PX** está apagando o primeiro caractere (**PX = 1**), volta-se para a linha 620, e o processo recomeça. Na linha seguinte, **PX** é decrementado de 1, pois estamos retornando a uma posição anterior. Verifica-se então se o cursor não caiu sobre um dos caracteres fixos do formato (cujas posições são representadas por **P1** e **P2**). Em caso afirmativo, recua-se o cursor mais uma posição. Na linha 648, o conteúdo de **W$** (onde a entrada está sendo montada) é atualizado e retoma-se o ciclo.

Não sendo <**CR**> ou ←, o caractere digitado é interpretado como uma entrada de dado. As linhas 634 a 638 checam se ela é válida. Para valores de **PP** iguais a 1, 2 e 6 (3 e 4 não são usados), apenas números são admitidos. Para **PP** igual a 5 (entradas alfanuméricas) admitem-se letras maiúsculas e números. Se o caractere não for válido, a execução é devolvida à linha 628. Caso contrário, a linha 640 se encarrega de colocá-lo na tela e adicioná-lo à variável **W$**, incrementando, em seguida, **PH** e **PX**. Verifica-se então se o cursor foi para uma posição de caractere fixo, que deve ser saltado.

Após todos esses passos, retornamos à linha 624, que encerra a entrada quando o último caractere do dado é digitado, desviando a execução para a linha 650. Esta linha e a 652 voltam a impressão de tela para o modo **NORMAL** e apagam a parte não utilizada do formato. Nesse ponto, a rotina de entrada de dados está terminada. Segue-se a checagem da validade de datas e horas por meio de operações lógicas.

APERFEIÇOAMENTOS

Podemos agora fazer algumas modificações na rotina — por exemplo, transferir a declaração da variável **WW$** da linha 601 para o início do programa, visto que só é necessário defini-la uma vez, e não sempre que a rotina for utilizada. Talvez você não precise usar todos os tipos de entrada. Nesse caso, é possível retirar as partes dispensáveis da rotina, pois elas são independentes. Com isso, economiza-se um pouco de memória.

Também é interessante fazer com que a tabulação vertical seja controlada, de modo que se possam entrar valores com mais de 39 caracteres.

```
10  HOME
20  PRINT  TAB( 8)"DEMONSTRACAO
 DE ENTRADA"; TAB( 55)"DE DADOS
"
30  VTAB 6: PRINT "SENHA DE ACE
SSO: ";:P$ = "G,05": GOSUB 600:
 IF W$ <  > "SENHA" THEN  END
40  VTAB 8: PRINT "DATA DE HOJE
: ";:P$ = "C": GOSUB 600:DT$ =
W$
50  VTAB 10: PRINT "HORA: ";:P$
 = "H": GOSUB 600:HR$ = W$
60  VTAB 12: PRINT "DIGITE UM V
ALOR: ": VTAB 13: HTAB 6:P$ = "
D10,3": GOSUB 600:VA =  VAL (W$
)
70  PRINT : PRINT DT$: PRINT HR
$: PRINT VA
80  END
596  REM  ********************
597  REM  *** SUB ENTRADA ***
598  REM  ********************
599  REM
600 PH =  PEEK (36) + 1:PP =  A
SC (P$) - 66
601 WW$ = "_____________________
_____": REM  * D
EFINA ESTA VARIAVEL NO INICIO D
O SEU PROGRAMA *
602  INVERSE
604  ON PP GOTO 606,608,0,0,612
,616
606 P = 8:PF = P:P1 = 2:P2 = 5:
WD$ = "DD/MM/AA":WC$ = "/": GOT
O 618
608 P =  VAL ( MID$ (P$,2,2)):D
 =  VAL ( RIGHT$ (P$,1)): IF D
= 0 THEN 614
610 PF = P:P1 = P - D - 1:P2 =
P1:WC$ = ".":WD$ =  LEFT$ (WW$,
P1) + WC$ +  LEFT$ (WW$,D): GOT
O 618
612 P =  VAL ( RIGHT$ (P$,2))
614 PF = 1:P1 = 99:P2 = P1:WD$
=  LEFT$ (WW$,P):WC$ = "": GOTO
 618
616 P = 5:PF = P:P1 = 2:P2 = P1
:WD$ = "HH:MM":WC$ = ":"
618  HTAB PH: PRINT WD$;
620 W$ = "":WI$ = ""
622 PX =  LEN (W$)
624  IF PX = P THEN 650
626 WA = 95 * (PP = 2 OR PP = 5
) + 68 * (PP = 1 AND PX < 2) +
77 * ((PP = 1 AND PX > 2 AND PX
 < 5) OR (PP = 6 AND PX > 2)) +
 65 * (PP = 1 AND PX > 5) + 72
* (PP = 6 AND PX < 2)
628  HTAB PH: PRINT  CHR$ (WA);
: HTAB PH: GET WI$
630 PA =  ASC (WI$): IF PA = 13
 AND PX >  = PF THEN 650
632  IF PA = 8 AND PX > 0 THEN
644
634  IF PP = 5 THEN 638
636  IF PA < 48 OR PA > 57 THEN
 628
638  IF PA < 32 OR PA > 127 THE
N 628
640  HTAB PH: PRINT  CHR$ (PA);
:W$ = W$ + WI$:PX = PX + 1:PH =
 PH + 1: IF PX = P1 OR PX = P2
THEN PX = PX + 1:PH = PH + 1:W$
 = W$ + WC$
642  GOTO 624
644 PH = PH - 1: IF PX = 1 THEN
 620
646 PX = PX - 1: IF PX = P1 OR
PX = P2 THEN PH = PH - 1:PX = P
X - 1
648 W$ =  LEFT$ (W$,PX): GOTO 6
26
650  NORMAL : FOR X = 1 TO P -
 LEN (W$)
652  PRINT  CHR$ (32);: NEXT
654  ON PP GOTO 656,666,0,0,666
,660
656 D =  VAL ( LEFT$ (W$,2)):M
=  VAL ( MID$ (W$,4,2)):A =  VA
L ( RIGHT$ (W$,2))
658 V = ((M = 4) + (M = 6) + (M
 = 9) + (M = 11)) * (D > 30) +
(M = 2) * (( INT (A / 4) = A /
4) * (D > 29) + ( INT (A / 4) <
  > (A / 4)) * (D > 28)) + ((A
< 1) + (M < 1) + (M > 12) + (D
< 1) + (D > 31)): GOTO 664
660 H =  VAL ( LEFT$ (W$,2)):M
=  VAL ( RIGHT$ (W$,2))
662 V = (H > 24) + (M > 59)
664  IF V THEN  PRINT  CHR$ (7)
;:PH = PH -  LEN (W$): INVERSE
: GOTO 618
666  PRINT : RETURN
```

UM EDITOR MUSICAL (1)

Transforme o microcomputador em uma partitura com composições musicais de sua própria autoria e, ainda, em uma orquestra que execute as melodias recém-criadas.

Existem duas maneiras de fazer o microcomputador tocar música. A primeira consiste em introduzir a melodia codificada em números dentro de um programa. A segunda, em utilizar um editor musical que cuide de toda a codificação, deixando ao usuário apenas a tarefa de digitar as notas.

Já vimos em *INPUT* como programar o micro para executar melodias. O processo é simples, mas requer um bom conhecimento dos comandos musicais do BASIC. Como estes nem sempre têm muita relação com música, o "compositor" é obrigado a consultar constantemente tabelas ou manuais, para verificar os valores adequados para a tonalidade, duração e volume, entre outros parâmetros de cada nota. A notação macromusical do MSX e do TRS-Color facilita esse processo, mas não elimina o trabalho com o restante do programa.

A execução de música através de programas apresenta outras desvantagens, como a dificuldade de editar a melodia e a impossibilidade de ouvir as notas antes que o programa esteja completo. Além disso, para cada peça temos que escrever um novo programa.

Um editor musical cuida de todos os detalhes da codificação, deixando o usuário livre para tratar da melodia propriamente dita. Não é preciso conhecimentos de programação para usá-lo.

Como os programas são bem longos, foram divididos em três partes. Digite agora a primeira parte e grave-a, para mais tarde completar a listagem.

COMO DIGITAR AS NOTAS

Uma vez que tenhamos o programa completo na memória do computador, poderemos selecionar as notas de várias maneiras. O processo será explicado pormenorizadamente para cada uma das máquinas no último artigo desta série.

Seja qual for o método escolhido, o usuário sempre terá a chance de alterar a melodia — mudando, inserindo ou apagando notas — até obter o resultado desejado. Se quiser conferir o trabalho, poderá executar a música — mesmo incompleta — a qualquer instante.

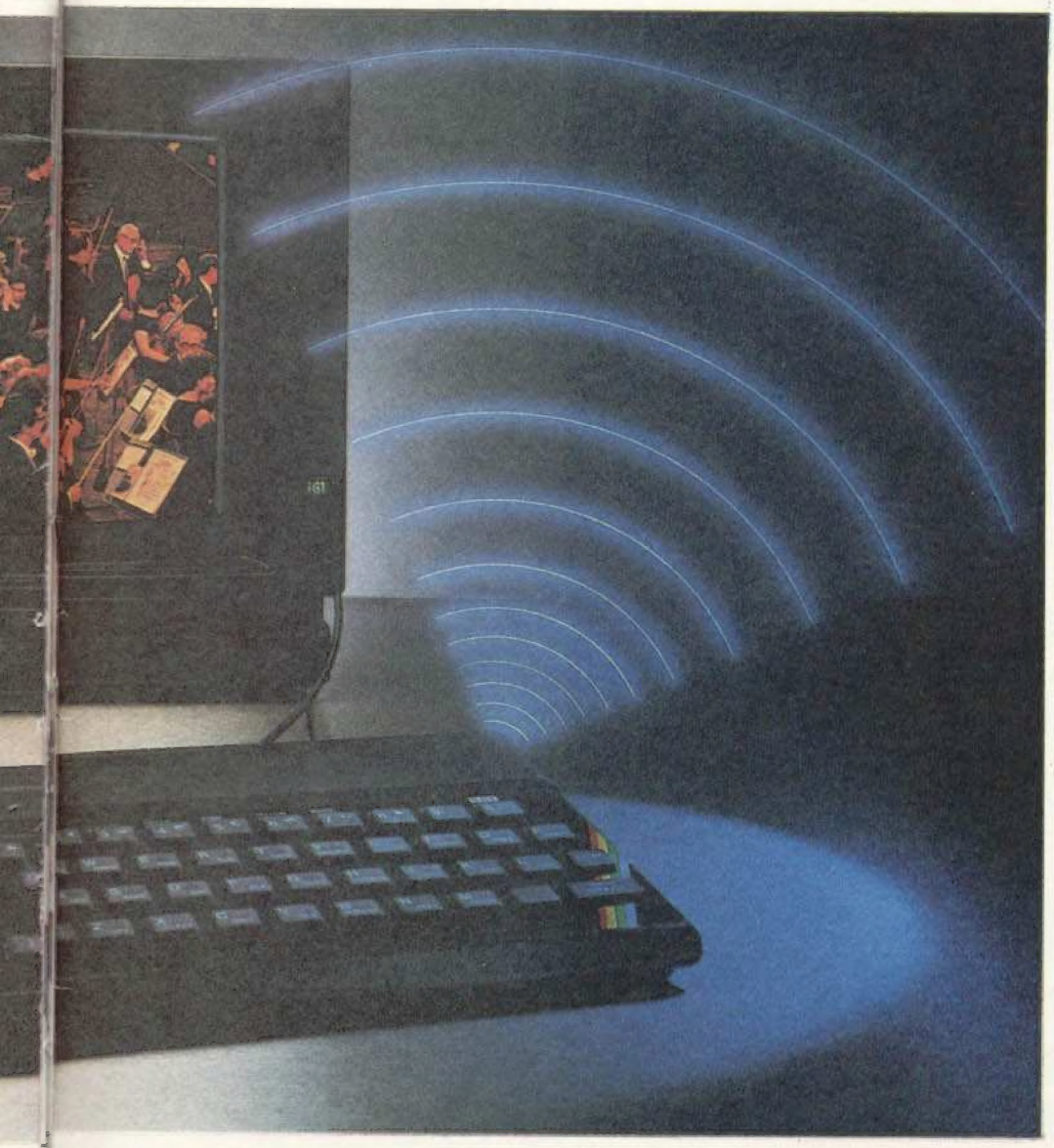
O editor musical permite ainda a mudança do andamento da oitava e, no caso do MSX, do timbre da nota.

As notas são selecionadas por meio de seus nomes (A, B, C, D...) ou do teclado, que é usado como se fosse um piano. Se o usuário preferir, poderá combinar os dois sistemas, introduzindo nota por nota em certos trechos e executando outros no teclado.

```
 10 BORDER 0: PAPER 0: INK 7:
 CLS
 40 LET maxnotes=1500: LET ct=
 0: LET tempo=.1: POKE 23609,
 128: POKE 23658,0: DIM m(35)
 95 FOR i=1 TO 35: READ m(i):
 NEXT i
100 DIM t(maxnotes+1): GOTO
 110
105 CLS : PRINT "Aguarde. Musi
ca sendo apagada."
110 FOR i=1 TO maxnotes: LET t
(i)=0: NEXT i
190 CLS
200 PRINT  INVERSE 1;AT 0,6;"
MENU PRINCIPAL "; INVERSE 0;''
''TAB 5;"[1] Tocar pelo teclado
"''TAB 5;"[2] Entrar  notas"''
TAB 5;"[3] Tocar de novo"
210 PRINT 'TAB 5;"[4] Editar m
usica"''TAB 5;"[5] Apagar musi
ca"''TAB 5;"[6] Salvar musica"
''TAB 5;"[7] Carregar musica"
270 PRINT ''TAB 5;"Opcao [S] p
ara sair";
300 LET A$=INKEY$: IF A$=""
THEN  GOTO 300
310 LET A=CODE (A$)
320 IF (A<49 OR A>55) AND A<>
113 AND A<>81 THEN  GOTO 190
330 IF A=49 THEN  GOSUB 1000
340 IF A=50 THEN  GOSUB 2000
350 IF A=51 THEN  GOSUB 3000
360 IF A=52 THEN  GOSUB 4000
365 IF A=53 THEN  GOTO 105
366 IF A=54 THEN  GOSUB 5000
367 IF A=55 THEN  GOSUB 6000
370 IF A=115 OR A=83 THEN
STOP
 380 GOTO 190
1000 CLS
1001 LET len=1: LET I$="Semicol
cheia"
1002 INPUT "[C]ONTINUE DA ULTIM
A NOTA    OU [I]NICIE NOVA MUSI
CA ? ";i$
1003 IF i$<>"c" AND i$<>"i" THE
N  GOTO 1002
1004 LET num=1: IF i$="c" THEN
 LET num=ct*2+1: LET tn=num
1005 IF i$="i" THEN  LET ct=0
1010 PRINT "Toque as notas nas
teclas de"'"<Q> - mais baixa
   ate"'"<M> - mais alta"'"Duas
 oitavas e meia estao dispo-niv
eis, sendo a tecla <I> o     C
medio"''''
1060 PRINT AT 17,0;"Duracao da
nota="
1070 FOR i=1 TO 100: NEXT i
1075 LET O$=""
1080 LET N$=INKEY$: IF N$="" TH
EN  GOTO 1075
```

- EXECUÇÃO DE MELODIAS ATRAVÉS DE UM PROGRAMA
- VANTAGENS DO EDITOR MUSICAL
- COMO DIGITAR AS NOTAS
- ALTERAÇÃO DA MELODIA
- CONFERÊNCIA
- COMPOSIÇÃO DE PEÇAS
- SELEÇÃO DE NOTAS

```
1085 IF N$=O$ THEN  GOTO 1080
1090 LET N=CODE (N$)
1091 IF N=33 THEN  LET len=1: L
ET I$="Semicolcheia"
1092 IF N=64 THEN  LET len=2: L
ET I$="Colcheia"
1093 IF N=35 THEN  LET len=4: L
ET I$="Seminima"
1094 IF N=36 THEN  LET len=8: L
ET I$="Minima"
1096 IF N=37 THEN  LET len=16:
LET I$="Semibreve"
1097 PRINT AT 17,16;"  ";I$;"
   "
1099 IF N<>32 AND N<>13 THEN  G
OTO 1109
1100 IF i$="i" THEN  LET ct=INT
 (num/2): RETURN
1102 LET ct=ct+INT ((num-tn)/2)
: RETURN
1109 IF N<60 THEN  LET index=N-
47
1110 IF N>90 THEN  LET index=N-
87
1112 IF N>=60 AND N<=90 THEN  G
OTO 1080
1115 IF index=5 OR index=9 OR i
ndex=16 OR index=20 OR index=21
 THEN  GOTO 1080
1116 IF index=2 THEN  LET t(num
)=len: LET t(num+1)=-4: GOTO 11
27
1118 IF index<=0 THEN  GOTO 108
0
1120 SOUND len/10,m(index)
1125 LET t(num)=len: LET t(num+
1)=m(index)
1126 PRINT AT 15,0;INT (num/2)+
1;" notas na memoria"
1127 LET num=num+2
1130 LET O$=N$
1140 GOTO 1080
```



```
10 CLEAR 5000:MX=250
20 R1$="whqestu":R2$="12345678"
30 R3$="abcdefgCDFGAp"
40 NY$="Q2W3ER5T6Y7UI900PZSXDCF
VBHNJM,L.Ç/[<"+CHR$(30)+CHR$(31
)+CHR$(29)+CHR$(28)+" "+CHR$(18
)+CHR$(127)+CHR$(13)+CHR$(11)
50 DIM N$(MX),C$(12)
60 FOR I=1TO7:READ LE$(I),L2$(I
):NEXT
70 FOR I=1TO12 :READ C$(I):NEXT
80 DATA SEMIBREVE,1,MINIMA,2,SE
MINIMA,4,COLCHEIA,8,SEMICOLCHEI
A,16,FUSA,32,SEMIFUSA,64
90 DATA c,C,d,D,e,f,F,g,G,a,A,b
100 FOR I=1 TO MX:N$(I)="****":
NEXT
110 NN=0: OC$="4":TE=120:LE$="w
" :LE=1
120 CLS:COLOR 15,2:KEY OFF:LOCA
TE 4,1:PRINT"COMPOSITOR MUSICAL
 MENU PRINCIPAL"
130 LOCATE 6,3:PRINT"1:RECUPERA
 MUSICA DA FITA"
140 LOCATE 6,4:PRINT"2:SALVA MU
SICA NA FITA"
150 LOCATE 6,5:PRINT"3:TOQUE NO
 TECLADO"
160 LOCATE 6,6:PRINT"4:COLOQUE
UMA TABELA DE NOTAS"
170 LOCATE 6,7:PRINT"5:MUDE O T
EMPO DE EXECUÇAO"
180 LOCATE 6,8:PRINT"6:LISTA/ED
ITA NOTAS"
190 LOCATE 6,9:PRINT"7:TOCA A M
USICA DA MEMORIA"
200 LOCATE 6,10:PRINT"8:MUDANÇA
 GERAL DE OITAVA"
210 LOCATE 6,11:PRINT"9:SAI DO
PROGRAMA"
220 LOCATE 5,19:PRINT"ENTRE COM
 O NUMERO DA OPÇAO";
230 A$=INKEY$ :IF A$<"1" OR A$>
"9" THEN230
```

```
240 OP=VAL(A$)
250 ON OP GOSUB 2080,2210,500,8
10,300,1510,1760,1910,270
260 GOTO 120
270 CLS:PRINT"VOCE TEM CERTEZA(
S/N)?"
280 A$=INKEY$:IF A$<>"S" AND A$
<>"N" THEN 280
290 IF A$="S" THEN CLS:COLOR 15
,4:END ELSE 120
300 CLS:COLOR 15,4:PRINT"MUDANÇ
A DO TEMPO DE EXECUÇAO"
310 PRINT:PRINT"TEMPO ATUAL:";T
E
320 PRINT"ENTRE NOVO VALOR DE T
EMPO";:INPUT ST
330 IF ST=0 THEN RETURN
340 IF ST<32 OR ST>255 THEN 320
350 TE=ST:RETURN
360 LOCATE 6,7:PRINT"NOTA    OI
TAVA    DURAÇAO"
370 U=9
380 FOR L=NN-5TONN
390 IF L<1 THEN 490
400 LOCATE 1,U:PRINT TAB(20);"
                ";
410 LOCATE 1,U:PRINT USING"###
";L;
420 A$=N$(L)
430 B$=LEFT$(A$,1):IF B$>="a" A
ND B$<="g" THEN C$=CHR$(ASC(B$)
-32)+" "ELSE C$=B$+"#"
440 IF B$="p" THEN C$="- "
450 PRINT "  ";C$;"       ";MID
$(A$,2,1);"       ";
460 PRINT LE$(INSTR(R1$,MID$(A$
,3,1)));
470 IF MID$(A$,4,1)="." THEN PR
INT"." ELSE PRINT
480 U=U+1:
490 NEXTL:RETURN
500 CLS:COLOR 15,4,15:C=VAL(OC$
):IF C>6 THEN C=6:OC$="6"
510 LOCATE 13,1:PRINT"MODO ORGA
O"
520 LOCATE 16,4:PRINT ;"
    ";
530 LOCATE 1,3:PRINT"OITAVA MAX
IMA:";C;TAB(46);"DURAÇAO:";LE$(
INSTR(R1$,LEFT$(LE$,1)));
540 IF MID$(LE$,2,1)="." THEN P
RINT"." ELSE PRINT
550 LOCATE 0,19:PRINT"---------
------------------------------"
;
560 LOCATE 0,20:PRINT" /  =OITA
VA       (INS/DEL) /  =DURAÇAO"
;
570 LOCATE 0,21:PRINT"ENTER =ME
NU                    CLS=APAGA"
;
580 GOSUB 360
590 IF NN=MX THEN LOCATE 1,18:P
RINT"MAXIMO NUMERO DE NOTAS COL
OCADO!";:FOR S=1TO3000:NEXT:RET
URN
600 I$=INKEY$:IF I$="" THEN 600
610 P=INSTR(NY$,I$):IF P=0 THEN
 600
620 IF P>36 THEN 650
630 OC$=MID$(STR$(C+INT((P-1)/1
2)),2):I$=C$(P-12*INT((P-1)/12)
)
640 GOTO 750
```

```
10 CLEAR 5000
20 MX=250:R1$="whqes":R2$="1234
5":R3$="abcdefgCDFGAp":NY$="Q2W
3ER5T6Y7UI900P@AZSXDCVGBHNMK,L.
"+CHR$(103)+"/^"+CHR$(10)+CHR$(
8)+CHR$(9)+" "+CHR$(21)+CHR$(93
)+CHR$(13)+CHR$(12)
30 DIM N$(MX),C$(12)
40 FOR I=1 TO 5:READ LE$(I),L2$
(I):NEXT
50 FOR I=1 TO 12:READ C$(I):NEX
T
60 DATA SEMIBREVE,1,MINIMA,2,SE
MINIMA,4,COLCHEIA,8,SEMICOLCHEI
A,16
70 DATA c,C,d,D,e,f,F,g,G,a,A,b
80 FOR I=1 TO MX:N$(I)="****":N
EXT
90 NN=0:OC$="3":TE=8:LE$="w ":L
E=1
100 CLS:PRINT@6,"COMPOSITOR MUS
ICAL":PRINT @40,"MENU PRINCIPAL
"
110 PRINT@96,"1:RECUPERA MUSICA
 DA FITA"
120 PRINT"2:SALVA MUSICA ATUAL
NA FITA"
130 PRINT"3:TOQUE NO TECLADO"
140 PRINT"4:COLOQUE UMA TABELA
DE NOTAS"
150 PRINT"5:MUDE O TEMPO DE DUR
ACAO"
160 PRINT"6:LISTA/EDITA NOTAS"
170 PRINT"7:TOCA A MUSICA DA ME
MORIA"
180 PRINT"8:MUDA A OITAVA GERAL
"
190 PRINT"9:SAI DO PROGRAMA"
200 PRINT@452,"ENTRE O NUMERO D
A OPCAO";
210 A$=INKEY$:IF A$<"1" OR A$>"
9" THEN 210
220 OP=VAL(A$)
230 ON OP GOSUB 1900,2020,480,7
60,420,1360,1560,1720,250
240 GOTO 100
250 CLS:PRINT" VOCE TEM CERTEZA
(S/N)?":POKE 282,255
260 A$=INKEY$:IF A$<>"S" AND A$
<>"N" THEN 260
270 IF A$="S" THEN CLS:END ELSE
100
280 PRINT@132,"NOTA  OITAVA   D
URACAO"
290 RT=0
300 FOR L=160 TO 320 STEP 32:PR
INT@L:NEXTL:PRINT@160,"";
310 FOR L=NN-5TO NN
320 IF L<1 THEN 410
330 PRINTUSING"###  ";L;
340 A$=N$(L)
350 B$=LEFT$(A$,1):IF B$>="a"AN
D B$<="g" THEN C$=CHR$(ASC(B$)-
32)+" "ELSE C$=B$+"#"
360 IF B$="p" THEN C$="- "
370 PRINT C$;"      ";MID$(A$,2,
1);"     ";
380 PRINT LE$(INSTR(R1$,MID$(A$
,3,1)));
```

MICRODICAS

MELHORE A QUALIDADE DO SOM

O programa apresentado neste artigo é bastante satisfatório para a composição e execução musical a nível amador. Os músicos profissionais, entretanto, logo notarão suas limitações — entre elas, a falta de recursos para a modulação do timbre (o que diferencia os vários instrumentos entre si quanto à sonoridade) e a impossibilidade de se obter efeitos polifônicos. Mas a culpa não é do programa.

Mesmo o MSX, o microcomputador mais sofisticado em termos de programação de efeitos sonoros, não é capaz de substituir um verdadeiro sintetizador musical. Este permite a simulação de um grande número de instrumentos de sopro, corda ou percussão, através da manipulação das formas de onda. Acoplando-o a seu microcomputador, por meio de uma interface MIDI (veja o artigo da página 1306), você poderá conseguir a qualidade máxima em matéria de som.

```
390 IF MID$(A$,4,1)="."THEN PRI
NT"."ELSE PRINT
400 IF RT<>0THEN RETURN
410 NEXT L:RETURN
420 CLS:PRINT@7,"MUDE OPCAO TEM
PO"
430 PRINT:PRINT"TEMPO ATUAL E";
TE
440 PRINT@128,"ENTRE NOVO VALOR
 DE TEMPO";:INPUT ST
450 IF ST=0 THEN RETURN
460 IF ST<00RST>255 THEN 440
470 TE=ST:RETURN
480 CLS:POKE282,255:C=VAL(OC$):
IF C>3 THEN C=3:OC$="3"
490 PRINT@7,"MODO ORGAO"
500 PRINT@64,"OITAVA:";C;TAB(10
);"DURACAO:";LE$(INSTR(R1$,LEFT
$(LE$,1)));
510 IF MID$(LE$,2,1)="." THEN P
RINT"." ELSE PRINT
520 PRINT@448,"C/B=OITAVA
ESQ/DR=DURACAO  ENTER=MENU
  CLEAR=APAGAR";
530 GOSUB 280
540 IF NN=MX THEN PRINT@416,"MA
XIMO NUMERO DE NOTAS COLOCADO!"
;:FORD=1TO1000:NEXT:RETURN
550 I$=INKEY$:IF I$="" THEN 550
560 P=INSTR(NY$,I$):IF P=0 THEN
 550
570 IF P>36 THEN 600
580 OC$=MID$(STR$(C+INT((P-1)/1
2)),2):I$=C$(P-12*INT((P-1)/12)
)
590 GOTO 700
```

CLASSIFICAÇÃO DOS MICROCOMPUTADORES POR LINHA

LINHA	FABRICANTE	MODELO
Apple II +	Appletronica	Thor 2010
Apple II +	CCE	MC-4000 Exato
Apple II +	CPA	Absolutus
Apple II +	CPA	Polaris
Apple II +	Digitus	DGT-AP
Apple II +	Dismac	D-8100
Apple II +	ENIAC	ENIAC II
Apple II +	Franklin	Franklin
Apple II +	Houston	Houston AP
Apple II +	Magnex	DM II
Apple II +	Maxitronica	MX-2001
Apple II +	Maxitronica	MX-48
Apple II +	Maxitronica	MX-64
Apple II +	Maxitronica	Maxitronic I
Apple II +	Microcraft	Craf II Plus
Apple II +	Milmar	Apple II Plus
Apple II +	Milmar	Apple Master
Apple II +	Milmar	Apple Senior
Apple II +	Omega	MC-400
Apple II +	Polymax	Maxxi
Apple II +	Polymax	Poly Plus
Apple II +	Spectrum	Microengenho I
Apple II +	Spectrum	Spectrum ed
Apple II +	Suporte	Venus II
Apple II +	Sycomig	SIC I
Apple II +	Unitron	AP II
Apple II +	Victor do Brasil	Elppa II Plus
Apple II +	Victor do Brasil	Elppa Jr.
Apple IIe	Microcraft	Craft IIe
Apple IIe	Microdigital	TK-3000 IIe
Apple IIe	Spectrum	Microengenho II
MSX	Gradiente	Expert GPC-1
MSX	Sharp	Hotbit HB-8000
Sinclair Spectrum	Microdigital	TK-90X
Sinclair Spectrum	Timex	Timex 2000
Sinclair ZX-81	Apply	Apply 300
Sinclair ZX-81	Engebras	AS-1000
Sinclair ZX-81	Filcres	NEZ-8000
Sinclair ZX-81	Microdigital	TK-82C
Sinclair ZX-81	Microdigital	TK-83
Sinclair ZX-81	Microdigital	TK-85
Sinclair ZX-81	Prologica	CP-200
Sinclair ZX-81	Ritas	Ringo R-470
Sinclair ZX-81	Timex	Timex 1000
Sinclair ZX-81	Timex	Timex 1500
TRS-80 Mod. I	Dismac	D-8000
TRS-80 Mod. I	Dismac	D-8001/2
TRS-80 Mod. I	LNW	LNW-80
TRS-80 Mod. I	Video Genie	Video Genie I
TRS-80 Mod.III	Digitus	DGT-100
TRS-80 Mod.III	Digitus	DGT-1000
TRS-80 Mod.III	Kemitron	Naja 800
TRS-80 Mod.III	Prologica	CP-300
TRS-80 Mod.III	Prologica	CP-500
TRS-80 Mod.III	Sysdata	Sysdata III
TRS-80 Mod.III	Sysdata	Sysdata Jr.
TRS-80 Mod.IV	Multix	MX-Compacto
TRS-80 Mod.IV	Sysdata	Sysdata IV
TRS-Color	Codimex	CS-6508
TRS-Color	Dynacom	MX-1600
TRS-Color	LZ	Color 64
TRS-Color	Microdigital	TKS-800
TRS-Color	Prologica	CP-400

CLASSIFICAÇÃO DOS MICROCOMPUTADORES POR NOME

FABRICANTE	MODELO	PAÍS	LINHA
Appletronica	Thor 2010	Brasil	Apple II +
Apply	Apply 300	Brasil	Sinclair ZX-81
CCE	MC-4000 Exato	Brasil	Apple II +
CPA	Absolutus	Brasil	Apple II +
CPA	Polaris	Brasil	Apple II +
Codimex	CS-6508	Brasil	TRS-Color
Digitus	DGT-100	Brasil	TRS-80 Mod.III
Digitus	DGT-1000	Brasil	TRS-80 Mod.III
Digitus	DGT-AP	Brasil	Apple II +
Dismac	D-8000	Brasil	TRS-80 Mod. I
Dismac	D-8001/2	Brasil	TRS-80 Mod. I
Dismac	D-8100	Brasil	Apple II +
Dynacom	MX-1600	Brasil	TRS-Color
ENIAC	ENIAC II	Brasil	Apple II +
Engebras	AS-1000	Brasil	Sinclair ZX-81
Filcres	NEZ-8000	Brasil	Sinclair ZX-81
Franklin	Franklin	USA	Apple II +
Gradiente	Expert GPC1	Brasil	MSX
Houston	Houston AP	Brasil	Apple II +
Kemitron	Naja 800	Brasil	TRS-80 Mod.III
LNW	LNW-80	USA	TRS-80 Mod. I
LZ	Color 64	Brasil	TRS-Color
Magnex	DM II	Brasil	Apple II +
Maxitronica	MX-2001	Brasil	Apple II +
Maxitronica	MX-48	Brasil	Apple II +
Maxitronica	MX-64	Brasil	Apple II +
Maxitronica	Maxitronic I	Brasil	Apple II +
Microcraft	Craft II Plus	Brasil	Apple II +
Microcraft	Caft IIe	Brasil	Apple IIe
Microdigital	TK-3000 IIe	Brasil	Apple IIe
Microdigital	TK-82C	Brasil	Sinclair ZX-81
Microdigital	TK-83	Brasil	Sinclair ZX-81
Microdigital	TK-85	Brasil	Sinclair ZX-81
Microdigital	TK-90X	Brasil	Sinclair Spectrum
Microdigital	TKS-800	Brasil	TRS-Color
Milmar	Apple II Plus	Brasil	Apple II +
Milmar	Apple Master	Brasil	Apple II +
Milmar	Apple Senior	Brasil	Apple II +
Multix	MX-Compacto	Brasil	TRS-80 Mod.IV
Omega	MC-400	Brasil	Apple II +
Polymax	Maxxi	Brasil	Apple II +
Polymax	Poly Plus	Brasil	Apple II +
Prologica	CP-200	Brasil	Sinclair ZX-81
Prologica	CP-300	Brasil	TRS-80 Mod.III
Prologica	CP-400	Brasil	TRS-Color
Prologica	CP-500	Brasil	TRS-80 Mod.III
Ritas	Ringo R-470	Brasil	Sinclair ZX-81
Sharp	Hotbit HB-8000	Brasil	MSX
Spectrum	Microengenho I	Brasil	Apple II +
Spectrum	Microengenho II	Brasil	Apple IIe
Spectrum	Spectrum ed	Brasil	Apple II +
Suporte	Venus II	Brasil	Apple II +
Sycomig	SIC I	Brasil	Apple II +
Sysdata	Sysdata III	Brasil	TRS-80 Mod.III
Sysdata	Sysdata IV	Brasil	TRS-80 Mod.IV
Sysdata	Sysdata Jr.	Brasil	TRS-80 Mod.III
Timex	Timex 1000	USA	Sinclair ZX-81
Timex	Timex 1500	USA	Sinclair ZX-81
Timex	Timex 2000	USA	Sinclair Spectrum
Unitron	AP II	Brasil	Apple II +
Victor do Brasil	Elppa II Plus	Brasil	Apple II +
Victor do Brasil	Elppa Jr.	Brasil	Apple II +
Video Genie	Video Genie I	USA	TRS-80 Mod. I

UM LOGOTIPO PARA CADA MODELO DE COMPUTADOR

INPUT foi especialmente projetado para microcomputadores compatíveis com as sete principais linhas existentes no mercado.
Os blocos de textos e listagens de programas aplicados apenas a determinadas linhas de micros podem ser identificados por meio dos seguintes símbolos:

Sinclair ZX-81

TRS-80

TK-2000

MSX

Spectrum

TRS-Color

Apple II

Quando o emblema for seguido de uma faixa, então tanto o texto como os programas que se seguem passam a ser específicos para a linha indicada.

NO PRÓXIMO NÚMERO

PROGRAMAÇÃO BASIC

Truques com o teclado do TRS-80: desativação da tecla <**BREAK**>, auto-repetição, cursor piscante.

APLICAÇÕES

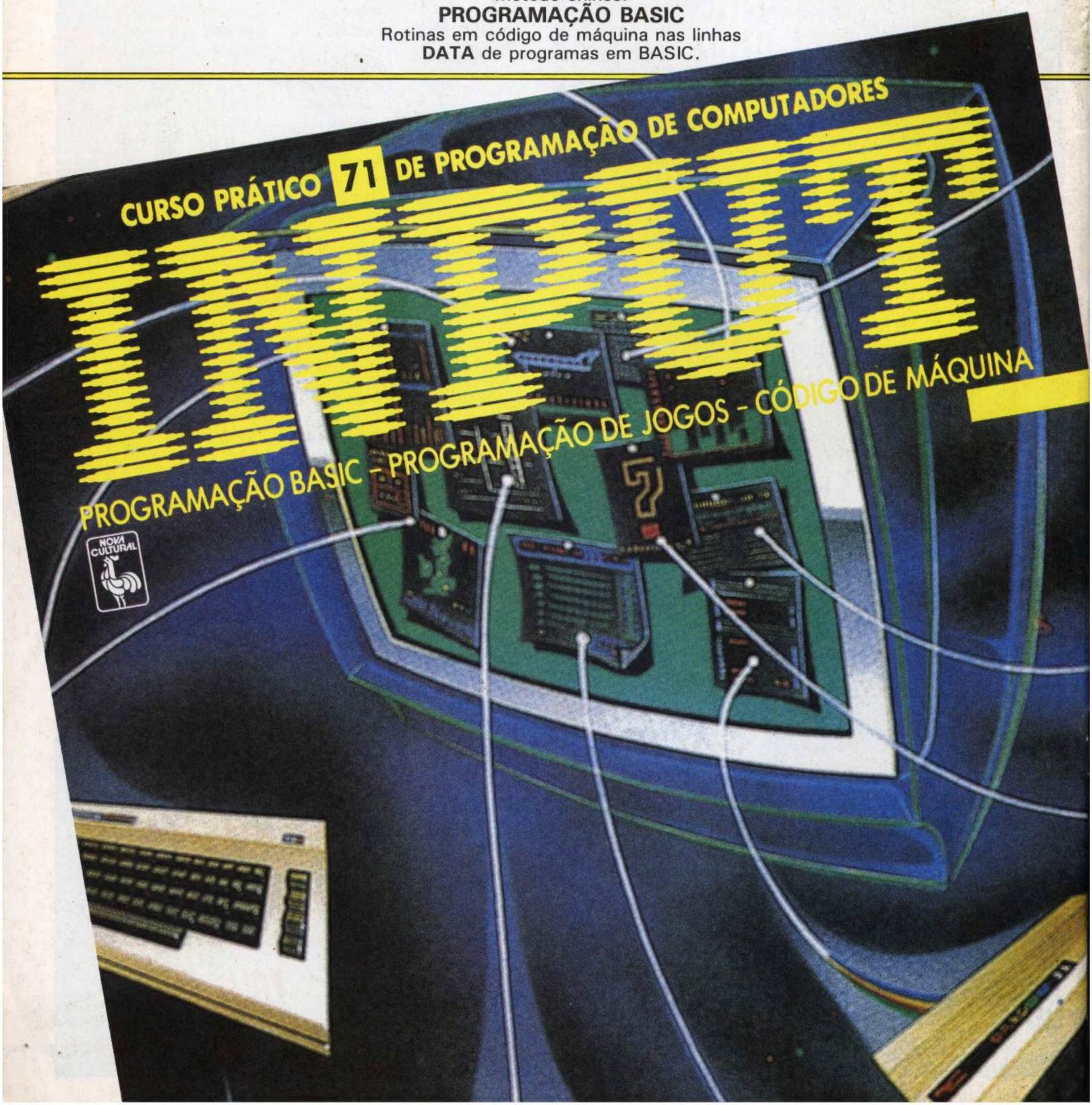
Digite a segunda parte do programa editor de melodias.

SOFTWARE

Quadro de avisos. Modem. Exploração da rede.

PROGRAMAÇÃO DE JOGOS

Um supercompressor de textos de aventuras. Método chinês.

PROGRAMAÇÃO BASIC

Rotinas em código de máquina nas linhas **DATA** de programas em BASIC.